

SyncWare News

The journal for technical
applications of T/S
computers

Volume 4 Number 5

May-Jun. '87

Fabulous Fest Feted

Mix 500 plus people who share a like interest with 52 vendors and displays, a superior hotel, service, and banquet, add over 116 door prizes and gifts, sprinkle in some terrific software and hardware bargains (Spectrum titles as low as 2 for \$5.00, new QL's with Psion software-\$99.00), plus some excellent speakers, and what do you get? The Second Annual Mid West Timex Computer Fest! If you missed this, you missed a really good time!

Mssers. Holmgren and Davis, and their staff, are to be commended for the fine job they did putting this year's Fest together. There's already talk about another Fest for next year in the Mid-west area, and possibly a small informal gathering for the West Coast area too. Stay tuned here for further developments!

Laptop Portable: 200 Pounds

By the time you're reading this, Cambridge Computer Ltd., Sir Clive Sinclair's new company, is expecting to be making deliveries of the Z88 Computer, according to Personal Computer World, Great Britain's prestigious microcomputer magazine. This new entry into the laptop computer market sports integrated software, a LCD display, tips the scales at 2 lbs., and has a 200 Pound price tag.

Some of the Z88's most notable features include: 128k ROM; RAM expandable to 3 Mbytes (it comes with 32K); permanent user storage in the form of plug-in EPROMs; RS232 port; ability to directly communicate with IBM compatible machines (the Z88 is not IBM compatible); bundled software consisting of a word-processor, spreadsheet, database, calculator, real-time clock, and alarm; a modem; and an 8 line X 100 character supertwist LCD display. It does not have a disk drive and apparently is also lacking any type of cassette interface.

At the time this article was written, Cambridge Computer Ltd. had not announced plans to market the Z88 in the United States. For more information, check out the March 1987 issue of Personal Computer World, or write Cambridge Computer Ltd., Cambridge CB4 1BR, U K.

Contest Closes

The Second Sort-of-Annual Programmer's Contest officially closed April 15th. Soon judging of the entries will be complete, the winners announced, and the prizes awarded. Be watching the July/August issue for the winning programs to be published!

SILICON MOUNTAIN COMPUTERS announces TRUE HIGH RESOLUTION SOFTWARE for the ZX81/TS1000. You read it right! Without any expensive hardware add-ons, your computer can now run software that even its designers never dreamed possible. Thanks to an amazing discovery by Wilf Rieger, and innovative programming by Gregory Harder and Fred Nachbaur, you no longer have to suffer the "low res blues." Multiple character sets, 256x192 graphics, 64-column screens, UDG's, even SPRITES are now available for your computer!

NO computer modifications are required. If you have a ZX81, TS1000, or TS1500, with a 16K (or larger) RAM pack, plus an 8K static RAM board, you already have all it takes to run this remarkable software. Suitable static RAMs include the popular "Hunter" board, or similar designs (see SyncWare News vol. 4 no. 1 for one such project costing under \$10). Alternately, consider our 8K "SCRAM" board (described below).

All prices include shipping in USA and Canada. Foreign orders: please add \$5 for air shipping. CDSs accepted at par from Canadian customers. Write for catalog of other available software. At SILICON MOUNTAIN COMPUTERS, the ZX01 family of computers is our ONLY specialty. Our goal is to develop the most progressive software ever created for these machines. We feel that the software listed below propels these machines into mainstream computing; we think that you'll agree.

With the loss of the "Hunter" board from the market, we saw the need for an improved functional equivalent, at a lower price! This board works with ZX81/TS1000 or TS1500, has on-board battery back-up protection, and supports all of our high-res software with no modifications. It can be mapped in 0-8K (ROM overlay), 8-16K (normal operation), even 16-24K or 24-32K.

Other features include:

- * DIP switch to deselect 2K blocks
- * Board enable switch
- * Write-protect switch
- * RESET switch easily installed (optional)
- * Very low power drain
- * Feed-through connector
- * FULLY ASSEMBLED! Just plug it in.
- * Use with other machine-code software

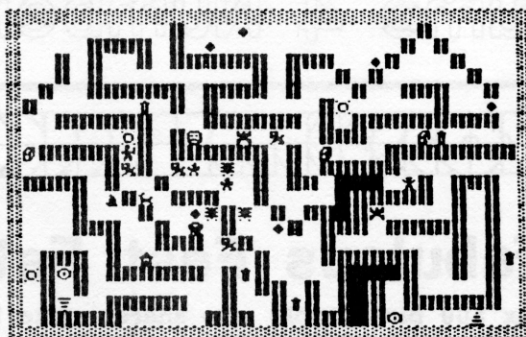
PRICE: \$39.95 including shipping.

DUNGEON OF YMIR

This D&D-style graphic adventure game is the most challenging, complex and spectacular entertainment software ever written for the ZX81 family of computers. 9 levels, 124 monsters (16 types), 5 spells, 15 objects. VERY addictive, VERY challenging, VERY impressive graphics. 100% machine-code, yet takes up all available memory in 15K. One reviewer calls it "5-star software" and "a must have." We know that you'll agree.

***** NEW LOWER PRICE *****

PRICE: \$19.95 including shipping



lit= 14 | [xp=4640 | co 0 | tel | sh2 | pr2 | rj0 | tex5

YEAR-AT-A-GLANCE

This calendar/appointment book program demonstrates the power you have using SRAM HI*RES Extended BASIC. Enter, update, delete, list, print messages and reminders for any day of any year 1800-2099. Program structure will remind you of much larger machines which shall remain nameless....

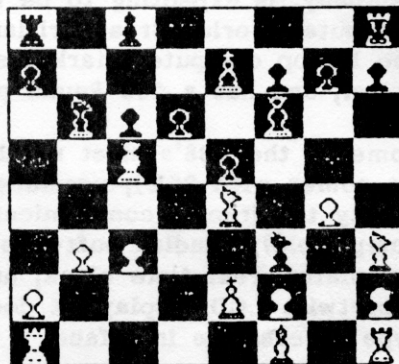
PRICE: \$9.25 including shipping.

*** SEE SPECIAL OFFER BELOW ***

[illegible]

YOU ZX-8

01	B2-B3	E7-E5
02	C1-A3	B8-C6
03	A3-F8	E8-F8
04	G1-H3	G4-G6
05	D2-D4	C6-D4
06	E2-E4	F6-E4
07	B1-A3	E4-C3
08	D1-D3	C3-D5
09	C2-C3	D8-F6
10	D3-C4	C7-C6
11	G2-G4	D4-F3
		CHECK
12	E1-E2	B7-B5
13	A3-B5	D5-B6
14	C4-C5	F8-E8
15	B5-D6	E8-E7
16	D6-E4	D7-D6
17		



If you purchase our "SCRAM" NVM board and SRA/ HI*RES EXTENDED BASIC and mention this ad, you'll get a FREE copy of YEAR-AT-A-GLANCE. Offer good only until July 1, 1987.

PRICE: \$24.95 incl. complete manual & shipping

The Peak of Quality... from SILICON MOUNTAIN COMPUTERS

C-12, Mtn. Stn. Group Box, Nelson BC V1L 5P1, CANADA (604)352-1668

FOR YOUR SUPPORT

This column announces any software, hardware, and related modifications that are new or otherwise untested by us. If you have something of interest, we will announce it here. Send us a description of your product (PLEASE keep it short). We advise readers to send a self addressed and stamped envelope, as a courtesy, to the individuals or companies for information.

AN-TO PRODUCTIONS, 9009 West Elm St. #2, Phoenix, AZ 85037, is currently marketing QUICKKEY 2068 keyboard overlays. Overlays are available for the following programs: AccuDraw T2, Artworx v1.1, Tasword/Tasprint, Mscript (regular and V5), and Omnicalc II. There is also a keyfinder and a blank overlay for your own programs. Price \$6.96 each.

BILL BELL, 596 Cherrington, Road, Westerville, Ohio 43081, (614) 882-3883, announces the availability of an Oliger Video Version of BBDOS for the TS1000 equipped with the Aerco FD-ZX Disk Interface and the JLO Video System. BBDOS is a fully automatic, BASIC transparent, disk operating system that locates itself in the 8-16K region of RAM. This BBDOS version enhances the operation of both the AERCO and JLO additions to the TS1000. It is fully compatible with all printer interfaces. Price \$29.95.

CHIA-CHI CHAO, 73 Sullivan Drive, Morgana, CA 94556, has his latest catalog out. Listed are many great buys on hardware and software. Software titles include works by such companies as ABBA Software and JRC Software. Both game and utilities are available. Many of which are available on 5.25" Aerco disk. A complete catalog is available for a SASE.

DAVID A. HILL, 1159 South Shore Drive #12, Holland, MI 49423, is publishing a newsletter for the users of AERCO's FD-68 Disk Controller. Published periodically, the newsletter contains a wealth of tips, program conversions for disk, and more. Send a SASE for more information.

FRED NACHBAUR, Silicon Mountain Computers, C-12, Mtn. Sta. Group Box, Nelson, BC V1L 5P1, Canada, has been hard at work under the mountain and has finally completed DUNGEON OF YMIR V3 for the ZX81/TS1000 with CMOS 8-16K RAM and 16K RAMPACK (64K Rampacks will also work provided the 8-16K region is disabled and the CMOS RAM connected). The circuit published in SWN 4:1 is adequate, as is the Hunter NVM (modified as per SWN 4:2). Price \$24.95 US.

KNIGHTED COMPUTERS, 707 Highland Street, Fulton, New York 13069, is selling QL computers, software, and related products. Their current flyer lists the book "TS2068 BASICS and BEYOND" by Sharon Z. Acker and the line of Intraclean Audio and Video Recorder care Systems among some of the products they are currently handling in addition to computer items.

C. W. ASSOCIATES, 419 N. Johnson St., Ada, OH 45810, (419)634-4874, are full line QL dealers. Contact them for your QL software and hardware needs. Items listed in the current flyer include an Expansion ROM Board, Front Page, Better Basic, Task Master, and games like Knight Flight, Karate, Squadrons, and Tank Blaster. MC/VISA now accepted.

RMG ENTERPRISES, 1419-1/2 7th Street, Oregon City, OR 97045, (503) 655-7484, announces the publication of a new book, ARCHIVE MASTER. This new work delves into the inner workings of the QL program Archive and contains over 100 tested procedures, tips, and tricks to help you make Archive an even more powerful working tool. Published in loose leaf format, it comes in its own three ring binder. Price \$44.95.

WEYMIL CORPORATION, Box 5904, Bellingham WA 98227-5904, announces NOVA 1000, a multi-tasking program for the TS1000. Features include a real-time clock, automatic program line tracer, auto repeat on all keys, and more. Price \$20.00. TRACER TS2068 is a debugging tool for BASIC programs. It allows you to see each line of your BASIC program as it is executed. Debugging has never been easier! Price \$15.00. SOUNDESIGN TS2068 is a sound development program for the TS2068. This program will allow you to manipulate the sound chip in ways you never thought possible. Once you have designed the sound you desire, a printout feature lets you obtain a hardcopy of the settings for use in other programs. Price \$15.00.

GREY & CLIFFORD COMPUTER PRODU S, P.O. Box 2186, Inglewood, CA 90305, (213) 759-7406, is offering software and hardware for communications on the Spectrum emulated TS2068. SPECTERM-64 is terminal emulation software which allows 64 column display without additional hardware. Price \$30. When used with the new Z-SI/O hardware card, it allows use with any RS-232 modem or peripheral including Hayes compatibles. Price for the Z-SI/O is \$79.95. Write for more details.

ROCKFORT PRODUCTS, 81 Church Road, London NW4 4DP, United Kingdom, announces the release of DISCIPLE Version 3 with 128K Snapshot Save. This new version will allow Disciple users to snapshot save the full 128K Spectrum memory to disk. Disciple is a multi-purpose Spectrum interface. For further information, contact Alan Miles at Rockfort Products.

CHARLES STELDING, 1415 South Baxter, Tyler, TX 75701 announces the availability of the TIMEX 2068 DESK TOP PUBLISHER. This new program allows the user to layout pages for bulletins, reports, announcements, programs, etc., with 'Paste-Up' capabilities. This program will generate headlines, text, and allow placement of SCREEN\$ anywhere on the page. The program allows the user to design his own fonts to add to the ones already included. This program will run the 2040 printer plus most other dot matrix printers. Cost is \$19.95 + S&H. A version is also available for the Olivetti 2300 for only \$24.95 + S&H. Look for a review of this program in a forth coming issue of SWN.

JOHN MCMICHAEL, 1710 Palmer Dr., Laramie, WY 82070, is offering a hardware/software package designed to interface the TS2068 to a Commodore 1520 Four-color printer/plotter. Prices range from \$14.95 ppd for the bare board and software, to \$30.95 for an assembled and tested unit. The software may be purchased separately for \$8.95 ppd. Write for details.

G. RUSSELL Electronics, RD1 Box 539, Centre Hall, PA 16828, (814) 364-1325, has NUCLEON (\$34.95) and TASKMASTER in stock. They are also offering some special sales on QL software. Spectrum ROMs for the TS2068 are currently available for only \$16.95. The May 1987 has also just been released. Call or write for details or their current catalog.

FORUM

Re-Runs Wanted

Dear Editor, As a newcomer to your magazine, having seen only two issues thus far, may I make the following suggestion? When you have the space, could you publish the index of your back issues? This would enable newcomers like myself to purchase back issues with articles of interest. Also, I have briefly seen a book with the title SyncWare News Volume 1. Would it be possible to publish the index, price, and ordering information if it is still available?

Gerd Breunung
B.C., Canada

Gerd- A comprehensive index of the back issues of SWN might be of interest to many people. Unfortunately, such a list is not available at this time, nor would it fit in this column. Be looking for it in a future issue of SWN.

Yes, SyncWare News Volume 1 is still available. It deals exclusively with the ZX81/TS1000 computer and contains Lprint Hints, Reviews, a Data Acquisition Program, Tyd-byts, a Math Development Program, Hardware Projects, and much more! The complete, bound, volume #1 is available from Thomas B. Woods, P.O. Box 64, Jefferson, NH 03583 for \$16.95. Non-US residents, please write for special ordering instructions.-ed.

Hi-Res Debate

Dear Editor, The idea of high resolution graphics for the ZX81 (see SWN 11/86, pg.13) is amazing even if not new. The program I bought, SW-HI-RES by Nissim Elmaleh is now 3 years old. It draws beautiful curves, scrolls, can generate a character set and probably more if I spent some time with it. It also works well, at least the demo program looks good, without ANY hardware modifications! Unfortunately, it will not print out through the Memotech Centronics I/F. Until it or Wilf Rigter's program can do that they are both useless as far as I am concerned. Apparently there are people who are satisfied with a pretty screen and printout to a T/S printer.

R. Arthur Gindin
Monroe, WI

Arthur- I agree that WRX16 is amazing, but disagree with your comment that it is "not new." WRX16 is TRUE high resolution. "SW-HI-RES" and at least four other similar packages are what I prefer to call "quasi-hi-res." Exact bit patterns are not possible with this approach. Furthermore, because of the extensive data manipulations required, it is also comparatively slow.

Getting "big printer" copy of quasi-hi-res screens is very involved, if not impossible. WRX16, however, uses a true bit-map, so getting screen dumps to your big printer via the Memotech (or any other) interface is simplicity itself. Proceed as follows:

1. BREAK operation of your program when the desired screen has been built up. NEW the computer.

2. LOAD Peter McMullin's big-printer version of SINC-ARTIST. (Available from Peter McMullin, 2340 Queen St. East, Toronto, Ontario M4E 1G9, Canada.) Configure it for your interface and printer. (NOTE: This works only with C. Itoh or Epson compatibles, but NOT Gorilla Banana because it only uses 7 vertical bits.)

3. Escape to BASIC from SINC-ARTIST using the SAVE function followed by ENTER only.

4. Transfer the contents of the high-res display file (8192-14336) into A\$ (which has been previously DIMensioned, ie. DIM A\$(6144).) This can be done by placing an LDIR uploader somewhere in low memory, by using the high memory version of HOT 2, or even with a simple BASIC PEEK loop such as:

```
100 LET S=8192
110 FOR N=1 TO 6144
120 LET A$(N)=CHR$(PEEK S)
130 LET S=S+1
140 NEXT N
```

5. Return to SINC-ARTIST using GOTO 6. Go to the print menu and dump your screen to your big printer.

SyncWare

Thomas B. Woods
Jeffrey D. Moore
Fred A. Nachbaur
Thomas J. Bent

Publisher
Editor
Technical Director
Asst. Editor

All subscription information, billing, and ad artwork should be addressed to:

SyncWare News
PO Box 64
Jefferson, NH 03583
(603) 586-7734

Article submissions, Forum, Support listings, Classified ads, and really easy questions should be addressed to:

SyncWare News
603 S. Mill St.
Louisville, OH 44641
(216) 875-1257

Article submissions, hardware, and everything else, especially the tough questions, should be addressed to:

SyncWare News
C-12, Mtn. Sta. Group box
Nelson, BC V1L 5P1
(604) 354-3858

Back issues of SWN are available for \$3.50 each. Volume #1 (for the 1000) is available for \$16.95.

SWN is done entirely on TS computers. Submissions are preferred as word processed text files recorded on good quality tape. Memotext for the 1000 and Mscript for the 2068 are the preferred word processors. Documents submitted in Tasword Two and other word processors must be done in 52 column format. If you do not have a word processor, but you feel you have a good idea, then don't hesitate too work it up and submit it. Complete writers guidelines are available upon request from the Ohio Editorial office. SWN pays authors about \$40.00 per page for original articles.

See my "DUNGEON OF VMIR" ad, and other upcoming ads, for actual examples of printouts made using this approach. I have available "THE HI-RES PRIMER FOR THE ZX81" that covers many questions regarding hi-resolution on the ZX81/TS1000. Readers interested in a copy of this booklet may order it directly from me at SILICON Mountain.-Fred Nachbaur

User Group Forming

Calling all Wisconsin, Illinois, and Iowa ZXers... Bill Heberlein is interested in forming a Timex User Group in his area. Others with similar interest should contact him at:

Bill Heberlein
5052 N. 91st Street
Milwaukee, WI 53225

A Day Late, A Dollar Short

Dear Editor, Thank you for your kind review of my book entitled Assembler Routines for the Z80. Unfortunately, the publishers, Century Communications Ltd., after a merger with Hutchinson, decided to cease publication of all computing books last August and remaindered all stock to Nigeria. Furthermore, I have what attempts I could to search out any available remaining books that may have been gathering dust on the shelves of computing bookshops, but to no avail.

I would, therefore, be grateful if you could print a suitable apology on my behalf in the first available issue in order to avoid disappointment to your readers.

David Barrow
West Yorkshire, Great Britain

Editor's note: While the book is no longer available, Mr. Barrow has given us permission to reprint certain portions of it. If there is suitable interest in us doing this, we will consider it for future issues of SWN.

ZX Clone?!

There is currently being marketed a ZX81/TS1000 clone under the name of the PC8300. Several readers have sent us info and/or inquiries about this 'new' computer.

Based on information we've gathered so far (we haven't actually had our hands on one yet), this is what we've learned.

It is an import from the Far East.

It is being marketed in this country through mostly surplus vendors for anywhere from \$19.95 and up.

It features an internal 2K of RAM, a real mechanical keyboard, reverse video, 8K ROM BASIC/OP System, Z80A @ 6.5 MHz, graphics, sound, TV or monitor mode, joystick IF, and cassette data storage. Also included are the necessary power pack, cassette, and video cables.

RAM may be upgraded to ? (assume 64K max.) with RAMPACK Upgrades which are purchased separately.

A Color Pack is available to allow use with either a color TV or monitor.

The ads claim the computer will run all pre-recorded tapes for the Sinclair ZX81 and Timex TS1000.

We have been told that if a ZX81/TS1000 program makes use of any ROM calls, the program is NOT compatible with the PC8300. Hence, it's possible that your MC programs either won't work at all, or will need to be rewritten.

We're currently testing this unit at Silicon Mountain, and will have a complete report as soon as possible. Any readers having any information about the PC8300 they'd like to share may do so by sending it to the Ohio Editorial Office.

2068 Space Bar

Dear Editor, If you're keeping a list of names of people who have had trouble with the TS2068 space bar, add mine. However, I've found another way to live with the problem.

Since I graduated directly from the ZX81 to the TS2068, I automatically started using the BREAK key instead of the space bar. It works like a charm...

Basil Wentworth
Bloomington, IN

Bug Alert

Dear Editor, Thank you for publishing my review of WINKJET I in SWN 4:4. However, somehow the phrase crept in "Included is a program called Typalot..."

Typalot must be ordered as a separate program from WIDJUP Co. Please advise your readers accordingly.

Robert D. Hartung
Huntertown, IN

Robert- Oops, my face is red! I checked my source disk and the manuscript was correct. I suppose during one of those late night editing sessions, as I nodded off at the keyboard, my fingers decided to lead a merry life of their own. My apologies to the WIDJUP Co., you, and any readers I may have inconvenienced.-jm

Replacement Power

Dear Editor, Through some experimentation which I won't go into here, my TS2040 printer power transformer was destroyed. Do you know where I can replace it? Radio Shack carries a transformer (273-1512) rated at 25.2 volts at 2.0 amps. Is that close enough to the 24V at 1.2 A of the original? Is there another place I could try? Should I try Alphacom?

William Frank
Benwood, WV

William- There's nothing 'special' about the voltage regulator in the TS2040 printer. It has a large heat sink, so it should be able to handle the slight overvoltage the Radio Shack transformer will deliver to it.

E.A. Brown, G. Russell Electronics, or Zebra may have an exact replacement transformer if you want to try them. However, it will probably be for a premium price.-jm

Display Trouble

Dear Editor, Ever since the 2068 first came out, I have tried to get a monitor that would work correctly with it. I started with a \$600 Mitsubishi. The top 2 to 4 character lines in the monitor mode tore to the left. Nobody knew what to do, including Mitsubishi. Over the years, I have tried several others. All of them had the same problem. After reading in SyncWare News, SUM, and the CATS Newsletter about the Sears TV/Monitor and the Magnavox 80 monitor, I decided to try again.

I installed a CMOS RGB circuit and proceeded to buy the Sears TV/Monitor. The same problem occurred in both the monitor AND the RGB mode. To insure the problem wasn't in my computer, I even tried another TS2068. The results were the same.

I tried adjusting VR1 in my computer and found that I could almost eliminate the tearing problem with the 10K pot set to the full 10K ohms.

"Aha!", I said, "If 10K ohms is good, then 20K ohms should be better."

I chipped out the 10K pot and installed a 20K pot in its place. Even with the pot set to the full 20K, the tearing continued to occur. I took back the Sears unit and purchased a Magnavox 80. Once again we encountered the same problem. It isn't quite so bad in the monitor mode (actually, it's almost acceptable), but the RGB mode is terrible.

Perhaps this might prove to be a clue to the problem. When I get a fairly good screen in the monitor mode, it's usually with a yellow or lighter colored border. A blue border shows the beginning of tearing, and a black border is much worse.

T. G. Morley
T or C, NM

Mr. Morley- The problem you describe is common with the TS2068's in composite video (or television). Though I haven't actually delved into it in great detail (I personally prefer the 'Oligerized' color TS1000), the problem appears to be because the sync level is too near the black level. As a result, it takes the monitor awhile to 'realize' that there are actually horizontal sync pulses there at all, and lock into them.

You shouldn't have trouble with this, however, using a good RGB interface. The reason for this is that the sync and video are separated for RGB. If you're considering upgrading to a disk drive, consider the AERCO which also has an RGB driver onboard. I'm not sure why your CMOS RGB driver isn't working, all I can suggest is to find a way to get a stronger horizontal sync signal.

If you insist on sticking with composite video, just about your only recourse is to avoid dark backgrounds, and carefully adjust the horizontal hold to reduce or eliminate the tearing. On some sets, reducing contrast (and brightness if required) helps.

We'll see that your letter is published. Perhaps one of our readers has something to add.-fn

Misc.

Dear Editor, I have several questions that perhaps you and your readers can provide answers for.

1) Although the Ramex disk system did not become the surviving disk system for the 2068, some of us still have it and would appreciate any support you could give, if nothing more than finding out who else owns the system. The biggest drawback I see with this system is its inability to allow bank switching, the use of cartridges, or memory expansion. If anybody has any fix for these problems, it would be nice to know about it.

2) After experience with VisaCalc, Appleworks, and Lotus 123, one of the most frustrating parts of TS2068 computing is the lack of a credible spreadsheet. Because of this, I wrote my own. However, since it is in BASIC, it is not exactly a Ferrari! I thought I had found the solution to my speed problems in the publication of TiMachine. However, a few problems exist:

a. It is not compatible with my DOS. A problem which can be worked out with a little creative programming.

b. The author of the program choose to leave out a runtime for VAL. Since most of my programs use strings for data storage compression, the compiler is virtually useless without major overhauls of the programs.

c. The compiled version uses its own variable table and cannot read the BASIC variable table so no sharing of data between BASIC and MC can take place without duplication and messy transfer between tables.

Are the problems mentioned above inherent in compilers or are they just the author's choice? Do you know of any solutions?

3) Is there a way to create new BASIC lines while in BASIC? I would like to write a complete BASIC line (including line number, etc.) in a string, transfer it to the area which checks syntax, add it to the proper location in BASIC, and return to the program without messing up the system variables irretrievably.

4) Two more routines would make the TS2068 truly useful, especially in business applications. Do you know if anyone has come up with a PRINT USING routine for right and left justification or a KILL routine to eliminate unneeded variables from memory to conserve space and to trigger the error trapping?

Thanks for your help.

Lyle Luzum
Calmar, IA

Lyle- Some of the questions regarding BASIC routines are really interesting. Alas, this job affords me very little time to 'play' at the keyboard.

Of all the compilers for the 2068 we've seen, TiMachine is probably the best of the lot. Even though a major rewrite of your program might be necessary, the features of TiMachine are worth it. Just remember, you don't have to compile everything. A routine to get around the VAL, while long and slow in BASIC, is nothing once compiled.

Here's your letter. Maybe some of our readers will come up with the routines you're seeking and improve TS2068 BASIC for all of us.-jm

If Wishes Were Horses...

Dear Editor, I have the Oliger Disk system and I think it's great, BUT...there are times when I 'wish' that somebody would announce the availability of their NEW PRODUCT: TS2068 HARD DISK SYSTEM.

John Constantine
Chapel Hill, NC

Romswitch & Big Printers

Dear Editor, I have a TS2068 with a ROM Switch for Spectrum, a Tasman CPI, and an Epson RX80 printer. I also have some pretty good programs for the Spectrum mode that would be terrific if there was some way to get them to print out on the Epson instead of just the TS2040 printer.

Anyone know of a big printer driver for the Spectrum mode TS2068?

George Cary
Coloma, CA

George- I contacted G. Russell Electronics, the best known ROM Switch producer to emulate the Spectrum on the TS2068. They said they believed there was software available, but were unable to give me a title. May I suggest Curry Computer or Peter McMullin?-jm

PC Findex From Tom Woods

(This is not Timex/Sinclair related so isn't in the 'For Your Support Column'.-ed.)

PRO/FILE, the popular data base program for ZX/TS computers has been refitted for use on PC compatible machines, says Tom Woods, PRO/FILE's creator. The new program, called FINDEX, is a much upgraded version which makes uses of the PC's disk drives, 80 column video, and greater memory capacity. New features include the ability to merge FINDEX data into other word processed documents (such as form letters). FINDEX can transfer data files from other computers (ie., Timex) into its data base through an RS232 serial printer interface. Data capacity is a whopping 131,000 bytes per file. Up to 4 files may be stored on a pair of disks--100 files on a 10 meg

hard disk! The program is being marketed as shareware for \$16.95. Contact Thomas B. Woods, PO Box 64, Jefferson, NH 03583 (603-586-7734) for more details and order info.

Not A Bug

But annoying none-the-less. Our printer got the listings for "ABC/123" nice and dark, but pushed the inverse graphics into the mud. Since some of these are crucial to proper program operation, here are these lines in a different format. Significant spaces are indicated with a crosshatch "#", and inverse video is underlined>.

```

7000 REM HEX LOADER/EDITOR
7090 PRINT AT 21,0;"CHECKSUM=";S
UM;"#OK? Y/N#####
10 REM DONT CHANGE ANYTHING
#BEFORE LINE NO. 10#
100 REM PAUSE
150 REM CLS SCROLL
200 REM REWARD
300 REM SCORE
500 REM HAPPY FACE
600 REM SAD FACE
700 REM ERASE FACE
800 REM WIPE
920 REM FLASH
1000 KEYBOARD CHARACTERS
1340 IF USR BT THEN REM <>0##TRY
##
1360 IF USR BT THEN REM <>0##AGAI
N#
2000 REM ADDITION
3000 REM SET UP TEST STRING
3300 REM SCRAMBLE TS
4000 REM ALPHA-SPACE BRIDGE
8800 REM DRILL COMPLETE
8880 IF USR BT THEN REM <>0(graphic on 5)x4,
(gr. on 8)x4
9000 REM INITIALIZE
9000 LET X$="* * * * *
9190 IF USR BT THEN REM <>0ABC/
123
9220 PRINT AT 8,5;"PARENT/TEACHE
R#SECTION","WHAT IS PUPILS NAME?
(TO 8 CHAR.)
9290 PRINT AT 14+VAL F$,N4;">"
9470 PRINT AT 22,0;"***PRESS#ANY
#KEY#EXCEPT#BREAK***"
9630 IF USR BT THEN REM <>0HELLO
#
9995 REM (gr. 5)ABC/123 BY F.NACHBAUR(gr. 8)
9997 REM (gr. 5)SW-LO-RES BY G.HARDER(gr. 8)

```

LOOK AT THIS!

Our 32K Non-Volatile RAM for the TS2068 has been re-designed, Improved, AND CUT IN PRICE!

32K (expandable to 64K)

was \$109.95

→ NOW \$ 75 (cheap!) ←

- * Fits in the cartridge slot
- * 32K expanded memory
- * Battery backed
- * Change battery without losing RAM
- * Manual write-protect switch
- * Create "Instant-On" software
- * Run your BASIC programs like a cartridge
- * Learn the secrets of Bank Switching
- * Just plug in another chip for 64K

Thomas B. Woods, P.O. Box 64, Jefferson, NH 03583

Soapbox

W. C. McGrath

After seeing some of the new software available for the ZX81/TS1000, specifically "Dungeon of Ymir" and "SRAM HI*RES Extended BASIC," something has occurred to me about the TS1000 lacking color and sound.

Color and sound is nice, and it can even be spectacular in a good game or whatever. However, I have noticed that after having had lots of it (Atari 2600, etc.), it simply becomes less important.

I think that one of the things that messed up the Atari games machines, is that a bunch of folks jumped in and made a lot of "games" which just rang the bells and flashed the colors and left very little of the (limited) memory for the GAME. Most Atari games, the GAME part, are garbage... rip-offs. They don't "call" anybody back to play again, not even children. And all the color and sound doesn't change that.

Also, apart from the rip-offs, there is simply the "saturation" matter. We've seen the color and heard the sound. It's not new anymore. We finally realize that it doesn't make a classic. Once everybody has it, it doesn't even give status anymore.

In art, color has its own special things that it can do. Effects that can be obtained in no other way and we wouldn't even want to detract from that; it is simply true. BUT NOTE that the same holds true for a black and white drawing. A fine charcoal on paper does not apologize to anybody. It has its own special effect. It is not intrinsically better or worse than color. It is just simply what it is, and it can be very fine. It has its own silvery dignity, its own moods.

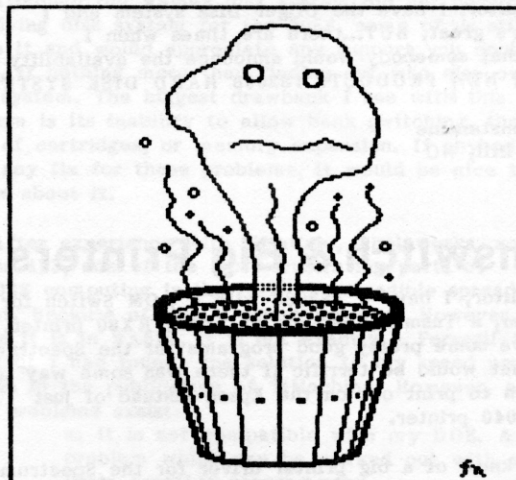
I don't even want to detract from the chunky stock graphics of the TS1000. It is what it is. But I've got to say that this new WRX16 high-res stuff that I've been seeing opens up a whole new world of moods and delicacy.

I've enjoyed "Ymir." I liked it at first, AND it keeps calling me back. That's the good part. It's a good GAME, not just the graphics. It holds up. Got time? Take it slow and cautious. Feel adventurous? I do the "Thou art most truly a fool" version, to see how far I can get with just Level 1 experience. If "Dungeon of Ymir" is not a classic, then I don't know what it is.

Who needs color and sound?



VAPORWARE



... From SILICON MOUNTAIN

Now that our SCRAM board is available and off the "vaporware" lists, we felt a need for new entries. We're presently taking orders for the following "new" products which definitely will maybe appear one day.

NERD-WORD

- The word-processor for the compleat idiot. Revolutionary input routine obsoletes the mouse, joystick, light pen, trackball, and even the keyboard! You don't even have to know the QWERTY Keyboard! Alphabet and control icons continuously scroll across the MCWP (Master Control Panel Window). When the desired character is in the CSF (Cursor Selection Field), press the button on our DBI (Doorbell Button Interface) (only \$235.90).

DEBASED

- The file manager of the future! Uses the same input routine as NERD-WORD. Amazing new sort routine; simply print out your files, cut them up, and sort by hand (free tutorial included). Then use our Laser Wand (only \$695.87) to read the sorted files back in!

MAIL-PURGE

- Interactively links NERD-WORD and DEBASED. Comes with a pair of scissors and a glue stick for inserting your datafiles into text files. Our patented KAMIKAZE routine eliminates the need for a DELETE function; when the program figures that you've had your files long enough, it assumes that you're tired of them and automatically purges them without saving. Even wipes out the program and gives you an op-art crash-screen!

BUGS-PUNNY

- The ULTIMATE computer game. You are supplied with 16K of pure machine-code; however, the bytes are completely scrambled. Your challenge is to arrange them into a working program. WE don't even know what the final program does! Comes with a packet of instant coffee, a box of double-stuf Oreos, and a pocket bottle of Extra-Strength Excedrin. We'll even PAY YOU \$10 if you send us a working copy of the solution!

BUT SERIOUSLY...

We DO have several innovative software titles for your ZX81, TS1000 or TS1500. Send two IRC's for free catalog.

SILICON MOUNTAIN COMPUTERS: C-12, Mtn. Stn. Group Box
Nelson, BC V1L 5P1 CANADA

Put Your Inkjet in Plotter Mode

OLIVETTI PR2300 "PLOTTER" MODE

Here is an answer to Robert Haley's letter in FORUM, SWN 4/2. I suspect the chief problem is understanding the PR2300 User Guide. It seems to have been written in Italian and translated into UK English (rather than American English) by someone whose first tongue was NOT English.

Awkward User's Guide notwithstanding, the PR2300 has some things to its credit. It uses ordinary paper, it's not expensive, and it has bit-mapped graphic capabilities.

According to the PR2300 User Guide there are two types of PR2300 firmware. I presume Mr. Haley has FWM 001, the regular ISO ASCII printer with plotter/graphics features.

Background

First of all, the PR2300 is NOT a plotter in the sense normally associated with plotting. It will not function directly as an X-Y plotter. The "PLOTTER PROGRAMMING CODES (FWM 001)" described in chapter 11 of PR2300 User Guide are the same as the "GRAPHIC MODE (FWM 001)" in chapter 6. THE PR2300 will print bit-mapped dots. This is the basis for the hi-res screen copy routines in commercial software. This feature may be used to "plot" bit-mapped data.

The PR2300 enters the graphics mode when it receives an ESC G. This command disables the PR2300's character generator and must be followed by several parameters which define the manner in which the data will be printed.

The first is a number from 1 to 255 which tells the PR2300 which dot column to start in. In other words, this defines the left-hand margin. A semicolon must follow the margin parameter.

The second is a number from 1 to 110 which defines the width of the map in 8-bit bytes. A semicolon must follow this parameter also.

Third is a number from 1 to 255 which defines the vertical dimension of the map in 8-line rows. This parameter is optional to the PR2300 processor. If this is omitted, the bit map will be only one dot row having a length defined in the second parameter. If this parameter is expressed and followed by the fourth parameter, they must be separated by a semicolon. If this parameter is omitted, the last parameter must also be omitted.

The fourth and last parameter is also optional. It may be a number from 1 to 9 and defines the vertical resolution. Its default is 2 which gives vertical dot size equal to horizontal dot size. I like to set this to 3 for copying text. It makes it more readable for me.

The parameters must be terminated with ESC Z.

The control codes and parameters must be sent in the language understood by the PR2300, i.e. ISO ASCII. The references above to ESC G, and ESC Z mean the

ASCII values for those characters must be sent to the printer. The parameters are not numerical values, but are the ASCII codes for the digits of the numbers. Leading 0's in the parameters may be omitted. I recommend fully expressing all the parameters to make the data easier to understand and change. The table shows this better than words.

The graphics mode control code may be preceded by either the zoom command (ESC /) or reverse image command (ESC -) or both. Zoom doubles the image horizontally and vertically. Reverse prints a negative image.

Here is an example of the command-code sequence to set up a map for a zoom image of 32 bytes horizontally by 24 bytes vertically (a ZX/TS screen map). The margin of 192 centers a zoom image on 80 column paper.

ASCII Hex Decimal

ESC	1B	27	Zoom
/	2F	47	Image.
ESC	1B	27	Graphic map
G	47	71	info to follow.
1	31	49	Left
9	39	57	margin
2	32	50	=192.
;	3B	59	Separator.
0	30	48	Map horiz
3	33	51	dimension =
2	32	50	32 8-bit bytes.
;	3B	59	Separator.
0	30	48	Map vert
2	32	50	dimension =
4	34	52	24 8-line rows.
;	3B	59	Separator.
2	32	50	Normal vert resolution.
ESC	1B	27	End of command
Z	5A	90	code sequence.

Please Note: These are NOT ZX-81/TS1000 character codes. ZX/TS alpha codes can be translated to ASCII by adding 27 dec; number codes by adding 20 dec.

The PR2300 will now fill the map with data bytes. The map is filled from left to right, top to bottom. Bytes are mapped MSB to LSB from left to right. When the map is full, the PR2300 returns to the normal mode with no further commands necessary.

The printer must be sent the exact number of bytes as defined in the command-code sequence. If too few are sent, the printer buffer may not be emptied. Too many will result in garbage being printed when the PR2300 returns to normal mode.

In the ZX/TS screen copy map described above, there are 32 horizontal bytes/line X 8 lines/row X 24 rows. A total of 6144 bytes are required to fill the map.

The manner of transmitting the data to the PR2300 depends on the user's system. For a memory-mapped

printer port, the data could be sent with POKES. For I/O-mapping, extended-BASIC OUT commands would work. A printer driver utility may allow the user to send the data by LPRINTing a string containing the command-code sequence or bits to be plotted, however, a machine code routine is probably the best method regardless of the type of port.

The BASIC

The included BASIC program allows the user to specify the location of the machine code routine, whether the printer port is I/O or memory mapped, the port status-read and data-write addresses, and margin size for screen copies. It also asks for the dimensions of the plot map in 8-bit columns and 8-line rows.

The program displays the calling addresses for zoom and normal copies and the addresses for plotter set-up and plotting, and asks for verification. Upon a positive response, the program will move RAMTOP if requested, create a REM if required, put the machine code in place, make a zoom copy of the final screen if the PR2300 is on line, and remove the BASIC program.

The BASIC is only a means of loading the machine code and adapting it to the user's system. Hopefully, the user will have some means of saving just the code. This could be a tape utility such as Z-XLR8, my 2K-SDP system, or saving the code in a REM to be merged with programs by means of Delphic or Z-Tools, Etc.

Machine Code

Please Note: The disassembly listing will not work as shown. The code must be personalized to the users system and the parameters must be set. This is done by the BASIC program but could be accomplished with pokes to the code if it is already in place. The 3-byte sequence at STAT (printer status-read) must contain an IN A,(nn) instruction for an I/O mapped port or a LD A,(nnnn) instruction for a memory-mapped port. "nn" or "nnnn" is the address of the status port. Likewise, DATA (printer data-write) must be changed to OUT (nn),A or LD (nnnn),A. If a 2-byte IN or OUT instruction is used, the third space is simply left at "00". Other necessary changes are the 3 blank bytes right after NRMI and the two 3-byte blanks in PLOT. The first is the space for the left margin parameter for a screen copy. The second and third are the horizontal and vertical dimensions for the plot map, respectively.

The machine code presumes that the printer port reads BUSY on status bit 4 and PAPER EMPTY on status bit 1. It gives report code "D" for break and report code "P" for printer error (open lid, paper empty, etc). If the user's hardware does not read the PAPER EMPTY status bit, the six bytes of the code starting with ERR? may be replaced with zeros.

The machine code has four entry points. The first three are S-ZO (setup zoom), S-NR (setup normal), and S-PL (setup plot). By calling into the code at one of these points, the desired command-code sequence is transmitted to the PR2300. This is accomplished by loading an address offset value in to the BC pair and a 01 or 03 into the E register. The call to TFAS at OSET (offset) leaves the return address from the call submerged on the stack. It is brought to the surface, popped into the HL pair and added to BC. When the code gets to XCMD (transmit command-code sequence), the HL pair is pointing 1 byte below the start of the data string and the E

THE INVERSE GRAPHICS READ AS FOLLOWS:

```

100 REM KEY-IN
200 REM DATA$ TO CODE$
300 REM POKE CODE
400 REM DEC TO HEX TO C$
500 REM DEC TO DEC TO C$
1000 REM CODE LOCATION
1015 LET O$=" OLIVETTI PR2300
    COPY/PLOT
1100 REM PORT DESCRIPTION
1200 REM COMMAND PARAMETERS
1300 REM SET RAMTOP/MAKE REM
1400 REM CREATE CODE $STRING
1600 REM REMOVE BASIC
1700 REM CHECKSUM FAILURE

5 REM OLIVEPLOT (C) 1987
KENT E. COOK, BX 1031,
TERRE HAUTE, IN 47808
10 SAVE "OLIVEPLOT"
15 GOTO 1000
100 REM KEY-IN
105 PAUSE 4E4
110 LET I$=INKEY$
115 RETURN
200 REM DATA$ TO CODE$
205 LET C$=""
210 LET CSUM=0
215 FOR X=1 TO LEN D$-1 STEP 2
225 LET C$=C$+CHR$ (16*CODE D$(
X)+CODE D$(X+1)-476)
230 LET CSUM=CSUM+CODE C$(LEN C
$)
235 NEXT X
240 RETURN
300 REM POKE CODE
305 FOR X=0 TO LEN C$-1
310 POKE A+X,CODE C$(X+1)
315 NEXT X
320 RETURN
400 REM DEC TO HEX TO C$
405 LET H=(INT (X/256))
410 LET C$(C+1)=CHR$ (X-256*H)
415 LET C$(C+2)=CHR$ H
420 RETURN
500 REM DEC TO DEC TO C$
505 LET H=INT (X/100)
510 LET T=INT ((X-100*H)/10)
515 LET U=X-100*H-10*T
520 LET C$(C)=CHR$ (H+48)
525 LET C$(C+1)=CHR$ (T+48)
530 LET C$(C+2)=CHR$ (U+48)
535 RETURN
1000 REM CODE LOCATION
1005 FAST
1010 CLS
1015 LET O$=" OLIVETTI PR2300
PR2300
1020 PRINT O$,"PRESS:"
"1" TO PLACE CODE ABOVE RAMTO
P,""2" TO PLACE CODE IN
REM,""3" TO SPECIFY ADDR
ESS"
1025 GOSUB 100
1030 IF I$<"1" OR I$>"3" THEN GO
TO 1025
1035 LET RMTF=I$="1"
1040 LET REM=I$="2"
1045 IF I$="3" THEN GOTO 1060
1050 LET ADDR=(16514+REM)+((PEEK
16388+256*PEEK 16389-192)*RMTF)
1055 GOTO 1100
1060 CLS
1065 PRINT O$,"MEMORY LOCAT
ION FOR CODE ="
1070 INPUT ADDR
1100 REM PORT DESCRIPTION
1105 CLS
1110 PRINT O$,"PRESS:"
"1" FOR I/O MAPPED PORT
"2" FOR MEMORY MAPPED PORT"
1115 GOSUB 100
1120 IF I$<"1" OR I$>"2" THEN GO
TO 1115
1125 LET IO=I$="1"
1130 CLS
1135 PRINT O$,"I/O" AND IO;"M
EMORY" AND NOT IO;" MAPPED PRINT
ER PORT","STATUS PORT ADDRES
S ="

```


register has bit 0 set to indicate a command-code sequence. Bit 1 of the E register is set for screen copy and reset for plot setup.

The fourth entry point is X-SD (transmit SEED). Calling in at this point loads the least significant byte of SEED into the A register and loads the E reg. with zero, indicating a single byte is to be transmitted to the PR2300. This is intended to be used as a means of plotting a byte which may be calculated in BASIC and put into SEED with a RAND statement (See the example plot programs. Beware of RAND 0. It doesn't work. To be safe, add 256 to all values. The MSB is ignored anyway). This routine may also be used in the PR2300 normal mode to print an ASCII character by RANDing the character code and calling X-SD with USR.

By calculating the command-code address and using the E register to keep track of things in the above manner, I was able to write the code with only relative jumps and no calls within the code itself. The code will run without modification anywhere the user's system will execute machine code. The only stipulation is the data strings containing the command-code sequence must be kept in the same position relative to the code (I.E. Move everything together).

The first two entry points in the code lead to XDPL (transmit display file), the screen copy program, which works with either a collapsed or an expanded display file. It does require that the character dot patterns are in the normal location in the ROM. The program loops through the display columns and lines, getting the character codes from the display file and locating the dot patterns in the ROM. The dot pattern code is complemented for inverse characters and the bits are sent to the PR2300.

The third and fourth entry points are called to set up a plot map and send data to the map as described above. A full-sized map of 110 X 110 bytes contains 774,400 bit locations. The sine wave plot demonstration program, which calculates a value for every bit, took nearly sixteen hours to run.

I also have 32 and 64 column copy routines for the TS2068. These may be patched into commercial software, such as TECH-DRAW JR, which do not support the PR2300.

PLEASE NOTE: The author will supply a tape or EPROM versions of this program to interested individuals for a nominal cost. For more information, write:

Mr. Kent E. Cook
PO Box 1031
Terre Haute, IN 47808-1031

```
1140 INPUT STAT
1145 IF STAT<0 OR STAT>((255*IO)
+(65535*NOT IO)) THEN GOTO 1140
1150 PRINT STAT
1155 PRINT "DATA PORT ADDRESS"
=
1160 INPUT DATA
1165 IF DATA<0 OR DATA>((255*IO)
+(65535*NOT IO)) THEN GOTO 1160
1170 PRINT DATA
1200 REM "COMMAND SEQUENCE"
1205 PRINT "LEFT MARGIN SIZE"
=
1210 INPUT LMRG
1215 IF LMRG<1 OR LMRG>255 THEN
GOTO 1210
1220 PRINT LMRG
1225 PRINT "PLOT MAP WIDTH"
=
1230 INPUT HOR
1235 IF HOR<1 OR HOR>110 THEN GO
TO 1230
```

```
1240 PRINT HOR
1245 PRINT "PLOT MAP HEIGHT"
=
1250 INPUT VER
1255 IF VER<1 OR VER>110 THEN GO
TO 1250
1260 PRINT VER
1265 PRINT "ZOOM COPY = ""US
R "";ADDR;"""";""NORMAL COPY = ""U
SR "";ADDR+5;"""";""PLOT SETUP
= ""USR "";ADDR+12;"""";""PLOTTER
= ""USR "";ADDR+103;"""";""
K? (Y/N)
1270 GOSUB 100
1275 IF I$<>"Y" AND CODE I$<>118
THEN RUN 1000
1280 PRINT AT 20,0;"
"
1300 REM "SET REMPTOR MAP" REM
1305 LET A=16444
1310 IF NOT RMTPT THEN GOTO 1345
1315 LET D$="2A0C4401140FF1922044
02B363E2BF92B2B220240C37606"
1320 GOSUB 200
1325 LET FAIL=1300
1330 IF CSUM<>1529 THEN GOTO 170
0
1335 GOSUB 300
1340 RAND USR A
1345 IF NOT REM THEN GOTO 1400
1350 LET D$="01C600217D40CD9E093
E761297237723772336C223772336EA2
A294001C60009222940C37606"
1355 GOSUB 200
1360 LET FAIL=1350
1365 IF CSUM<>3023 THEN GOTO 170
0
1370 GOSUB 300
1375 RAND USR A
1400 REM "CREATE CODE STRING"
1405 LET D$="017E0018030180001E0
318050192001E01CDE7023B3BE109237
EB72052CB4B236F"
1410 LET D$=D$+"2A0C40231E020618
C50E00E5062056CB7228042B971837E5
6A2600CBB0292929D5"
1415 LET D$=D$+"11001E1916005919
D1CB7A7EE1281E2F18182310D8E10CCB
5928CFC1CB762328FB10C41829"
1420 LET D$=D$+"CDE7023A32401E00
F5CD460F3802CF0C000000CB4F2802CF
18CB6720ECF1000000CB43208CCB4B20
C4C30702"
1425 LET D$=D$+"1B2F1B470000003B
3033323B3032343B321B5A001B473030
313B0000003B0000003B321B5A000000
0000000000"
1430 GOSUB 200
1435 LET FAIL=1400
1440 IF CSUM<>13511 THEN GOTO 17
00
1445 LET H=INT (STAT/256)
1450 LET L=STAT-256*H
1455 LET C$(120)=CHR$ ((58*NOT I
O)+(219*IO))
1460 LET C$(121)=CHR$ L
1465 LET C$(122)=CHR$ (H*NOT IO)
1470 LET H=INT (DATA/256)
1475 LET L=DATA-256*H
1480 LET C$(134)=CHR$ ((50*NOT I
O)+(211 AND IO))
1485 LET C$(135)=CHR$ L
1490 LET C$(136)=CHR$ (H*NOT IO)
1495 LET X=LMRG
1500 LET C=152
1505 GOSUB 500
1510 LET X=HOR
1515 LET C=174
1520 GOSUB 500
1525 LET X=VER
1530 LET C=178
1535 GOSUB 500
1540 LET A=ADDR
1545 GOSUB 300
1600 REM "REMOVE REMPTOR"
1605 LET D$="2A0C40114341CD5D0AC
D9A14217D402229402A3240E9"
1610 GOSUB 200
1615 LET FAIL=1600
1620 IF CSUM<>1704 THEN GOTO 170
0
1625 IF REM THEN GOTO 1640
1630 LET C$(5)=CHR$ 125
1635 LET C$(6)=CHR$ 64
1640 LET A=16444
1645 GOSUB 300
1650 RAND ADDR
1655 RAND USR A
1700 REM "CHECKSUM FAILURE"
1705 CLS
1710 PRINT "CHECKSUM FAILURE","E
XAMINE D$ CHARACTERS",,,,
1715 LIST FAIL
```

```

7F40 017E00 S-ZO LD BC,007E
7F43 1003 JR SET3
7F45 01E000 S-NR LD BC,0080
7F48 1E003 SET3 LD E,03
7F4D 1005 JR OSET
7F4C 019200 S-PL LD BC,0092
7F4F 1E001 SET1 LD E,01
7F51 CDE702 OSET CALL TFAS
7F54 30 DEC SP
7F55 30 DEC SP
7F56 00 POP HL
7F57 00 ADD HL,BC
7F58 00 INC HL
7F59 00 LD A,(HL)
7F5A 00 OR A,XMTA
7F5B 20050 CNT? BIT 1,E
7F5D CB400 LD Z,EXIT
7F5F 20000 XDFL LD HL,(DFIL)
7F61 200C40 LD HL,INC HL
7F64 20 LD E,02
7F65 1E00 LD B,18
7F67 0618 ROWS PUSH BC
7F68 05 LD C,00
7F6A 0E00 LINE PUSH HL
7F6C 05 LD B,20
7F6D 0620 CLMN LD D,(HL)
7F6F 06 PAD? BIT 6,D
7F70 00072 JR Z,MULT
7F72 20000 PAD. DEC HL
7F74 00 SUB A
7F75 00 JR XMTA
7F76 1037 MULT PUSH HL
7F78 05 LD L,D
7F79 00 LD H,00
7F7A 20000 RES 7,L
7F7C CB00 ADD HL,HL
7F7E 00 ADD HL,HL
7F7F 00 ADD HL,HL
7F80 00 ADD HL,HL
7F81 00 PUSH DE
7F82 11001E LD DE,CTBL
7F83 10 ADD HL,DE
7F85 10 LD D,00
7F86 05 LD E,C
7F88 19 TABL ADD HL,DE
7F8A 01 POP DE
7F8B CB7A INV? BIT 7,D
7F8D 7E LD A,(HL)
7F8E E1 POP HL
7F8F 201E BITS JR Z,XMTA
7F91 2F INVT CPL
7F92 181B JR XMTA
7F94 23 INC HL
7F95 10D8 CLM? DUNZ,CLMN
7F97 E1 POP HL
7F98 0C LIN? INC C
7F99 CB59 BIT 5,C
7F9B 28CF JR Z,LINE
7F9D C1 POP BC
7F9E CB76 END? BIT 6,(HL)
7FA0 23 INC HL

```

```

7FA1 28FB JR Z,END?
7FA3 10C4 ROW? DUNZ,ROWS
7FA5 1028 JR EXIT
7FA7 CDE702 X-SD CALL TFAS
7FAA 3A3240 LD A,(SEED)
7FAD 1E000 SET0 LD E,00
7FAF F5 XMTA PUSH AF
7FB0 CD460F BRK? CALL BREK
7FB3 3802 JR C,STAT
7FB5 0F ERRO RST 08H
7FB6 00 Error D
7FB7 00 STAT NOP
7FB8 00 NOP
7FB9 00 NOP
7FBA 00 NOP
7FBB 00 ERR? BIT 1,A
7FBC 2002 JR Z,BZY?
7FBE 0F RST 08H
7FBF 13 ERRP Error P
7FC0 CB67 BZY? BIT 4,A
7FC2 200EC JR NZ,BRK?
7FC4 F1 POP AF
7FC5 00 DATA NOP
7FC6 00 NOP
7FC7 00 NOP
7FC8 CB43 CMD? BIT 0,E
7FCA 20030 JR NZ,XCMD
7FCB 00 DFL? BIT 1,E
7FCC 20044 JR NZ,XDF2
7FCE 00 EXIT JP MDCK
7FDF 0702 ZOOM DEC DE
7FE0 1B CPL
7FE1 2F DEC DE
7FE2 1B LD B,A
7FE3 00 NOP
7FE4 00 NOP
7FE5 00 NOP
7FE6 38 DEC SP
7FE7 3033 JR NC,8010
7FE8 303B30 LD (303B),A
7FE9 30343B LD (3034),A
7FEA 321B5A LD (5A1B),A
7FEB 00 NOP
7FEC 1B PLOT DEC DE
7FED 47 LD B,A
7FEE 3030 JR NC,801B
7FEF 313B00 LD SP,003B
7FF0 00 NOP
7FF1 00 NOP
7FF2 38 DEC SP
7FF3 00 NOP
7FF4 00 NOP
7FF5 321B5A LD (5A1B),A
7FF6 00 NOP
7FF7 00 NOP
7FF8 00 NOP
7FF9 00 NOP
7FFA 00 NOP
7FFB 00 NOP
7FFC 00 NOP
7FFD 00 NOP
7FFE 00 NOP
7FFF 00 NOP

```

```

456789ABCDEFGHIJKLMNPOQRSTUVWXYZ
RNDINKEY$PI????????????????????
????????????????????????????????
????????????????????????????????

```

```

*AT TAB ?CODE VAL LEN SI
N COS TAN ASN ACS ATN LN EXP INT
SGR SGN ABS PEEK USR STR$ CHR$
NOT ** OR AND (<=>=<=>) THEN TO STE
P LPRINT LLIST STOP SLOW FAST NE
U SCROLL CONT DIM REM FOR GOTO G
OSUB INPUT LOAD LIST LET PAUSE N
EXT POKE PRINT PLOT RUN SAVE RAN
D IF CLS UNPLOT CLEAR RETURN COP
Y

```

EXECUTED
 ON
 A "2K"
 MACHINE WITH
 COLLAPSED DISPLAY FILE
 ZOOM COPY

```

456789ABCDEFGHIJKLMNPOQRSTUVWXYZ
RNDINKEY$PI????????????????????
????????????????????????????????
????????????????????????????????

```

EXECUTED
 ON
 A "64K"
 MACHINE WITH
 EXPANDED DISPLAY FILE
 NORMAL COPY

Quick Print Your Pro/File & Tasword Files

by Robert D. Hartung

One day as I was drumming my fingers on the desk-top while waiting for my Tasword tape to load, I began thinking how ridiculous it was to be loading the whole program just to print out one page of a form or letter. All that I really needed was a printer driver and a short routine to read the file bytes and line-feeds into it. (If your I.F. has an onboard universal driver, you can skip the next paragraphs and adjust the listings accordingly.)

Since my I.F. requires loading in a software driver routine, I use the 117-byte routine in Tom Woods' ProFile for this and other programs like Beta Basic to output to a full-size printer. (To use the P/F printer routine in Spectrum mode, it is only necessary to POKE 63688,84 and POKE 63689,31 before relocating the code.) Any other relocatable driver routine could also be used.

To shorten the loading time as much as possible, I decided to put the printer code in a line 0 REM statement. I hear some "Not THAT again!" groans from those who recall painfully pecking in one character at a time from a ZX81 KBD, but really this is a very practical way of loading in one pass a short block of machine code along with the Basic.

The repeat-key function of the TS2068 makes an easy job of filling out the REM statement to the required length. Just key in 1 REM and then hold a key down until four lines are filled with any character. This will give a 122-character statement, which is a few bytes more than enough for the P/F routine but if you use another it may need a longer statement. Enter as line 10 LET R=PEEK 23635+256*PEEK 23636 and as line 20 POKE R+1,0. When you RUN this, line 1 will be changed to line 0 to prevent accidental editing or over-writing when it is merged with another listing.

Now enter LISTING 1, which loads the code from the second segment of ProFile and then copies it into the line 0 REM location. The pokes in line 30 place a block-move routine in the UDG location. This leaves all but a few hundred bytes of RAM clear for re-locating code almost anywhere, which makes this a handy utility program that you might want to save for other uses.

The origin for the printer routine in P/F is 63672, the number of bytes is 117, and the destination may be entered as the expression R+5. These values must be adjusted accordingly if any other routine is being used or if you use this utility program to move blocks of code to and from other locations in RAM. After the last value is entered and the code copied into 0 REM, that line will then show an error if you try to use LIST without a line number since it contains some non-character codes.

After the code has been copied into 0 REM, enter LISTING 2 beneath it. Note that the printer commands as given for a C. Itoh 8510 will need to be adjusted for the commands required by your printer. The page-header routine is optional and may be omitted

if you want to shorten the loading and printing times. Entering a "0" at the prompt allows inserting a new page whenever "0" is keyed during printing. When ENTER is keyed, this page is given a numbered header before proceeding with the printing. If any number other than "0" is entered during the initial prompts, the printing will stop automatically at multiples of that number, or else a form-feed command could be inserted here for your printer if desired.

```
0>REM .....
.....Long enough
to hold printer driver code....
5 CLEAR (PEEK 23627+PEEK 2362
8*256+200)
10 LET R=PEEK 23635+256*PEEK 2
3636
20 PRINT FLASH 1;"LOAD CODE to
be moved": LOAD "CODE": CLS
30 POKE 65368,1: POKE 65371,17
POKE 65374,33: POKE 65377,237:
POKE 65378,176: POKE 65379,201
40 INPUT "bottomADDR=": (PEEK 2
3730+PEEK 23731*256): topADDR=6
5367:"ORIG? ":or: BYTES? ":by:
DEST? ":ds: IF ds+by>=65367 THE
N PRINT FLASH 1;"DEST address to
o high": GO TO 50
50 POKE 65370,INT (by/256): PO
KE 65369,by-PEEK 65370*256: POKE
65373,INT (ds/256): POKE 65372,
ds-PEEK 65373*256: POKE 65376,IN
T (or/256): POKE 65375,or-PEEK 6
5376*256
60 RANDOMIZE USR 65368
70 CLS: INPUT "SAVE name?":n#
SAVE n#CODE ds,by
80 STOP
200 SAVE "MOVEC" LINE 5
```

Listing 1

If the graphic character CODE 138 is imbedded in your text as the first character in any line, it may be used to change fonts in single-spaced printing. The graphic character CODE 140 will change the following block of text from double to single-space in a different font, and inserting the graphic character CODE 143 into the line following the block will change them both back again. Note that these imbedded characters will have other functions in a normal Tasword printout and should be omitted in those text files. If you have no need of this feature or the insertion of a bit-space between characters, the routine will run faster with lines 9720 to 9730 and the bit-space command in line 9740 omitted. Printing skips first line Tasword graphics.

RUN 9999 to save this to tape as your abbreviated driver for Tasword files. When loaded, key ENTER to load a file into it. By ruthlessly editing prompts and using token-words wherever possible in the original Tasword listing, I also made room to merge this driver with it so I have the menu option of numbered page headers and variable bit-spacing in proportional mode when I want them. Since it is all in Basic, it is not as fast as the Tasword printout routine but works fine for mss. like this one.

```

0 REM Printer driver code
9000:ON ERR GO TO 9100
9100 CLEAR 33279
9200 ON ERR RESET : CLS : INPUT
"ENTER "" to LOAD new text fil
e ENTER any character to LPRINT
",a$: IF a$="" THEN PRINT #0; F
LASH 1;" LOADING TEXT CODE " : LO
AD ""CODE
9700 LET b=PEEK 23635+PEEK 23636
+256+5: LET i=INT (b/256): LET j
=b-i+256
9705 CLS : ON ERR GO TO 9800: LE
T c$="0000": POKE 26703,j: POKE
26704,i: LPRINT "1BP1B11B15
1BT11": LET b=1: INPUT "Page he
ader--N/L none",a$:"Dbl.sp? y/N
L":c$(1);"LINE NO. for auto-p
age OR 0?":c$(2 TO ): LET j0=VA
L c$(2 TO )
9708 LET c=j0: LET d$="(1+INT (N
-33344)/64)"
9710 FOR N=33344 TO 52480 STEP 6
4
9715 INPUT "": PRINT #0;AT 2,0;"
LINE "VAL d$;" Key 0 for next
page at line "VAL c$(2 TO )+c$
(j0=0):" BREAK LPRINT TO STOP "
9720 IF PEEK N=140 THEN LET c$(1
)="": LPRINT "1BE": NEXT N
9725 LET c$(1)="y" AND PEEK N=1
43: IF PEEK N=140 OR PEEK N=138
THEN LPRINT "1BP": NEXT n
9730 IF c$(1)="y" THEN LPRINT "1
BP"
9735 FOR K=N TO N+63
9740 PRINT CHR$ PEEK K: LPRINT
CHR$ PEEK K;"1B2":
9745 NEXT K
9750 LPRINT
9750 IF INKEY$="0" OR VAL d$=VAL
c$(2 TO ) THEN GO TO 9900
9770 NEXT N
9800 PRINT FLASH 1" END LPRINT
": PAUSE 100: LPRINT : GO TO 920
0
9900 LET c$(2 TO )=STR$ (j0+VAL
d$): LET b=b+1: INPUT "Insert PA
GE ":(b);b$: LPRINT "1BXPAGE "
b;"--":a$;"1BY": LPRINT : IF
c$(1)="" THEN LPRINT
9910 IF c=0 THEN LET c=VAL d$
9920 GO TO 9770
9999 SAVE "printAS" LINE 9000

```

Listing 2

If you do not have ProFile on a cartridge, you may at times want a quick printout of all or just the first part of a file as it was last ordered and saved to tape. It is a little more complex to create a similar routine to print P/F data, address files, etc., since the item lines are not all neatly stored in a 64-column field as they are in Tasword. However, the end-of-line and end-of-item codes (1 and 42) can easily be used to line-feed the printer.

First, the P/F listing is altered slightly as shown in LISTING 3 so that only those bytes from the part of the d\$ file actually in use are saved and loaded. If the value of P is noted for each file saved as code only, LOAD ""CODE can then be used to load it back into the identical P/F listing from which it was saved. Re-defining P with the value noted for that particular file will then allow restarting the P/F program with GO TO 1.

```

25 LET OR=PEEK 23627+256*PEEK
23628+26
107 IF X$="SAVE" THEN INPUT "CO
DE OR P/F? " :X$: IF X$="P/F" THE
N SAVE {# LINE 1: INPUT "VERIFY
Y/N? " :X$: IF X$(1)="Y" THEN VER
IFY {# : GO TO 1
108 IF X$="CODE" THEN SAVE {#CO
DE OR P-20: INPUT "VERIFY Y/N? "
:X$: IF X$(1)="Y" THEN VERIFY {#
CODE : GO TO 1

```

Listing 3

Such a file can also be loaded into another program, provided memory is cleared for it and the beginning address (OR) is defined. Note that no change can be made in the listing for ProFile from which the data is to be used in the printer routine without also making a corresponding change in the numeric value inserted for ORP/F in line 40 of LISTING 4. To do this, LET b = (the five-digit address) obtained by the direct command PRINT OR after stopping the P/F program during the main menu display.

```

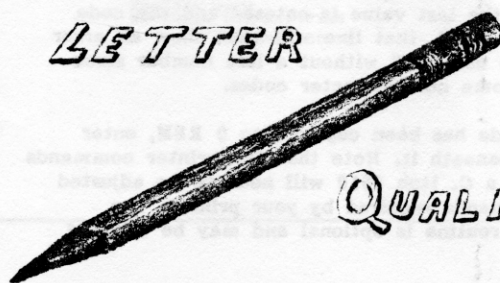
0 REM Printer driver code
10:ON ERR RESET : CLS : INPUT
"ENTER "" to LOAD new text fil
e ENTER any character to LPRINT
",a$: IF a$="" THEN CLEAR 30000
: PRINT #0;FLASH 1;" LOADING FI
LE": LOAD ""CODE
20 ON ERR GO TO 100: LET b=PEE
K 23635+PEEK 23636+256+5: LET I=
INT (b/256): LET L=b-I+256
30 CLS : POKE 26703,L: POKE 26
704,I: LPRINT "1BN1B11B15+1
BT11": INPUT "Page header--N/L n
one",a$
40 LET b=ORP/F: LET L=1: LET P
=2: LET J=1
50 INPUT "": PRINT #0,"LINE "
:L
70 IF PEEK b=42 OR PEEK b=1 TH
EN PRINT : LPRINT : LET b=b+1: L
ET L=L+1
80 PRINT CHR$ PEEK b: LPRINT
CHR$ PEEK b
90 LET b=b+1: GO TO 50
100 PRINT FLASH 1" END LPRINT
": PAUSE 100: LPRINT : GO TO 10
110 LPRINT : INPUT "Insert PAGE
":(P);b$: LPRINT "1BXPAGE "P;
"--":a$;"1BY": LPRINT : LPRIN
T : LET P=P+1: RETURN
120 REM Insert for envelopes:
60 IF PEEK b=42 THEN LPRINT :
INPUT "": INPUT FLASH 1;" NEXT e
nvelope " :b$
130 REM Insert for page headers
70 IF PEEK b=42 OR PEEK b=1 TH
EN PRINT : LPRINT : LET b=b+1: L
ET L=L+1: IF INKEY$="0" THEN GO
SUB 110
140 REM Insert for 64 columns:
70 IF J>=2 AND (PEEK b=42 OR P
EEK b=1) THEN PRINT : LPRINT : L
ET b=b+1: LET L=L+1: IF INKEY$="
0" THEN GO SUB 110
150 REM Insert for 64 columns:
75 LET J=J+(PEEK (b-1)=1): IF
PEEK (b-1)=42 THEN LET J=1
160 REM Insert to repeat any
memo (if first line in P/F item
is blank):
80 PRINT CHR$ PEEK b: LPRINT
CHR$ PEEK b: IF PEEK b=1 AND PE
EK (b+1)=1 THEN LPRINT "ATTENTIO
N OF:"
9998 ON ERR RESET : GO TO 100
9999 SAVE "printP/F" LINE 10

```

Listing 4

The REMs in LISTING 4 provide options for adding 64-column printouts and numbered page headers, and for stopping the printer to insert the next envelope to be addressed. An appropriate vertical tab command for your printer could be inserted in the listing to use address label rolls or envelopes with a fan-fold web for continuous feed instead of stopping.

LETTER



QUALITY

TS1000 Mystery Program

by Gregory C. Harder

"How about a 'Mystery Program' for SWN? This would be something really bizarre, yet interesting to a lot of readers. We would supply nothing but a listing; no documentation, no explanation. Let 'em figure it out themselves!" -tbw

"Sounds great. I know just the program. Greg Harder wrote this nifty _____ program, called _____. Only problem is, that it has an extensive BASIC listing with instructions, that would give it away. So, we could chop it down to the bare minimum to run the program. For those that never do figure it out, we could print the full BASIC portion in the subsequent issue. Greg, is it OK with you if we run your _____ in this fashion?" -fnn

"Sure, go for it. Sounds like fun." -gch

LISTING #1

```
1 REM "MEGA-REM", 1600 SPACES
  LONG. SEE BASILS COMPENDIUM
  SWN 4:1, OR OTHER ISSUES, OR
  USE A TOOLKIT.....
  .....
  PEEK 16511+256*PEEK 16512
  MUST RETURN 1602.

1200 REM HEX LOADER/EDITOR
1210 FAST
1220 CLS
1230 PRINT "START=";
1240 INPUT BASE
1250 LET SUM=0
1260 FOR L=1 TO 20
1270 GOSUB 1500
1280 NEXT L
1290 PRINT AT 21,0;"CHECKSUM=";S
    UM;" OK? Y/N      "
1300 INPUT Y$
1310 IF Y$="N" THEN GOTO 1400
1320 IF BASE<>17794 THEN PRINT A
1330 STOP
1400 PRINT AT 21,0;"WHICH LINE I
    S WRONG (1-20)?  "
1410 INPUT L
1420 GOSUB 1500
1430 LET SUM=0
1440 FOR N=BASE TO BASE+319
1450 LET SUM=SUM+PEEK N
1460 NEXT N
1470 GOTO 1290
1500 INPUT L$
1510 PRINT AT L-1,0;L$
1520 FOR N=0 TO 15
1530 LET V=16*(CODE L$(2*N+1)-28
    )+(CODE L$(2*N+2)-28)
1540 POKE BASE+16*(L-1)+N,V
1550 LET SUM=SUM+V
1560 NEXT N
1570 RETURN
```

So, here is the SWN "TS1000 Mystery Program." Enter a mega-REM (1600 bytes) and Listing 1. After entering each of the five screens of hexcode shown below, your checksum should agree with that printed onscreen; if not, find which line(s) have errors, and repair them before going on.

Then delete the loader lines, enter line 5 SLOW, LIST 5, and POKE 16419,5. (Why?) Save this to tape for next time. Then add the lines of Listing 2, which is the bare-bones BASIC required to run the program. Save this also. What IS this program? When will we run the explanation? Don't ask me, I've said too much already.... -fnn

HEXCODE LISTING

START = 16514

```
7676AF3222402A0C40062023368010FB
232336800615237EFE762804368818F6
2B36802323368010ED061F23368010FB
C92A0C40012B000901050B79E5C52336
8E10FB3D280AC10505E11122001918EC
C1E178192BE523368E10FB3D28050603
E118F0E179192B41E5368E2310FB3D28
03E118F1E1060378192BE523368E10FB
3D28050603E118F0E17906032323192B
2BE5C5368E2310FB3D2806C10404E118
EDC1E101F801ED423E05060BE5C51D36
8E1910FB3D2808C10505E11C1918EDC1
E1061423368E10FB0E7FED423E05060B
E5C5368E1910FB3D2809C10505E11D19
1C18EDC1E11919061A368E2B10FBED52
36921D1D193693017B01ED4236BB1C1C
061619368810FB36A6016D01ED420605
36882310FB2B2B3695C97EFE95C02A
5C4236973E0A112100010100E5093606
ED523606ED423603ED42368619368619
```

CHECKSUM = 27663

START = 16834

```
36060936830936860321210019EBF5C5
CD0C42C1F1E13D20D31E0B2A0C400618
2B237EFE76200410F81805C6807718F1
1D20E82A0C4001160309361C237E3DFE
3320043625180177AFC90100090B78B1
20FBC93E1EED47C9E53A344032334021
32407E07AE0F23862B77E1C9CD1A42E6
03C9CD1A42E60FC9CD1A42E60747CD1A
42E601803CA7C9CD1A42E607C9000000
00001C4C2A4C664D744D744D834DBE4E
CD1A42E601C92A0C4001230009225442
E5010E0009225642013C010922584201
0E0009225A42225C42010F0009225E42
013B0109226042E1CD3442FE05CA2D43
CD2E42FE012824FE02283FFE0328572A
58423612CD74453688CA684223CD9D41
C8EBCD0746EBCA684218E72A5642363B
CD74453688CA684201210009CD9D41C8
EBCD0746EBCA684218E42A5E423613CD
74453688CA68422BCD9D41C8EBCD0746
```

CHECKSUM = 28489

START = 17154

```
EBCA684218E72A60423626CD74453688
CA6842012100A7ED42CD9D41C8EBCD07
46EBCA684218E2000000003697225442
ED4B5642A7ED42E52025CD2E42FE0320
1E2A544201210009CD9D412002E1C9EB
CD0746280AEB363BCD7445368820E5E1
7DFE00200A2A5642012100092256422A
5442CD7445C8360823CD9D41C87EFE08
2808FE012804FE8020A1CD4942FE0220
1FE52A5642CD34422306004F097EFE80
280DFE08280936012A5C4209225E42E1
2B3697225442ED4B5E42A7ED42E52022
CD2E42FE03201B2A54422BCD9D412002
E1C9EBCD0746280AEB3613CD74453688
20E8E17DFE0020072A5E422B225E422A
5442CD7445C8360801210009CD9D41C8
7EFE082808FE872804FE8020A4CD4942
FE02202CE52A5E42CD3A424711210019
10FD7EFE802818FE0828143687A7ED52
ED5B5E42A7ED52EB2A5C4219226042E1
```

CHECKSUM = 31189

START = 17474

```
012100A7ED423697225442ED4B6042A7
ED42E52027CD2E42FE0320202A544201
2100A7ED42CD9D412002E1C9EBCD0746
280AEB3626CD7445368820E3E17DFE00
200C2A6042012100A7ED422260422A54
42CD7445C836082BCD9D41C87EFE0828
08FE022804FE80209DCD4942FE022027
E52A6042CD3442FE0F28F92B06004FA7
ED427EFE80280FFE08280B36022A5C42
A7ED42225842E1233697225442ED4B58
42A7ED42E52022CD2E42FE04201B2A54
4223CD9D412002E1C9EBCD0746280AEB
3612CD7445368820E8E17DFE0020072A
5842232258422A5442CD7445C8360801
2100A7ED42CD9D41C87EFE082808FE04
2804FE8020A2CD4942FE02202BE52A58
42CD3A4247112100ED5210FC7EFE8028
16FE082812360419ED5B5842A7ED52EB
2A5C4219225642E101210009C32D4300
0000E5EBCD0C422A5A427EFE95280236
```

CHECKSUM = 31403

START = 17794

```
88AFDBFEFE2F2018A7012100ED42225A
42CD074628087EFE88205C368C3CE1C9
FE3B201601210009225A42CD07462808
7EFE882042368C3CE1C9FE3D20132B22
5A42CD074628087EFE88202B368C3CE1
C9FE37201323225A42CD074628087EFE
882014368C3CE1C9ED5B5A422A5C42A7
ED522803EB36883C2A5C42225A42E1C9
00000000002A5A42E5A7ED52E1C00164
5A3606511520FD3686511520FD10F2CD
6242280236882A0C4001020309E57E3C
FE262803771805361C2B18F2E1606216
1C7EBA203B2B10F97E151506051414BA
280410F9182A011700097EFE25200736
342B361D1806FE3428023C773A0E42FE
01280D3D320E423A1246060A90321246
AF2A5C42225A42C90000002A0C400618
237EFE76200310F8C9EE807718F20000
00000000000000000000000017171717
171717171717171717171717171717
```

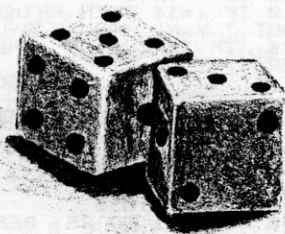
CHECKSUM = 24701

LISTING #2

(Enter lower case letters in the reverse {graphic} video mode.)

```
5 SLOW
20 LET DF=PEEK 16396+256*PEEK
16397
25 RAND USR 16516
190 RAND USR 18061
200 FOR N=1 TO 20
210 NEXT N
220 RAND USR 16516
230 RAND USR 16563
240 FOR N=1 TO 20
250 NEXT N
260 RAND USR 16516
270 PRINT AT 10,1;">>>>>FROM HA
RDER SOFTWARE.<<<<<"
290 FOR N=1 TO 40
300 NEXT N
340 CLS
380 RAND USR 16516
810 SLOW
850 LET B=9
860 POKE 16910,B
870 POKE 17938,B*10
880 PRINT AT 23,5;"press a key
when ready"
890 GOTO 890+(INKEY$="")
900 GOTO 900+(INKEY$<>"")
920 RAND USR 16516
930 PRINT AT 23,0;" 00000
10"
940 RAND USR 16563
950 FOR N=0 TO PEEK 16910+3
960 NEXT N
970 RAND USR 17000
980 IF PEEK (DF+791)=28 THEN GO
TO 1020
990 RAND USR 16516
1000 GOTO 940
1020 LET V=DF+767
1030 LET G$=CHR$ (PEEK V+128)+CH
R$ (PEEK (V+1)+128)+CHR$ (PEEK (
V+2)+128)+CHR$ (PEEK (V+3)+128)+
CHR$ (PEEK (V+4)+128)
1040 CLS
1050 RAND USR 16516
1060 PRINT AT 3,11;"end pgmo;AT
8,5;"final [ ";G$;AT 13,7;"spee
d [ ";CHR$ (PEEK 16910+156);AT 1
8,5;"run again!{y\n}"
1070 LET Y$=INKEY$
1080 GOTO 1070-290*(Y$="Y")+20*(
Y$="N")
1100 CLS
1110 FOR N=0 TO 30
1120 POKE 16918,N
1130 RAND USR 16917
1140 NEXT N
1150 STOP
9995 REM *****
9996 REM * SWN MYSTERY PGM. *
9997 REM * G.C. HARDER *
9998 REM * SEPT 1984 *
9999 REM *****
```


YAHTZEE



by ART MARZANO

Yahtzee is a poker-style dice game where each player tries to roll combinations to match the categories on the scorecard. The program is set up for 4 players. There are 5 dice. All are automatically rolled on each first roll. Each player is allowed 3 rolls per category. There are 13 categories.

The object of the game is to fill in all of the categories on the scorecard while accumulating as many points as possible. All categories **MUST** be filled in even if this means scoring a ZERO. The highest possible score is 375 (I don't think anyone has ever done that).

The two-section scorecard is set up as:

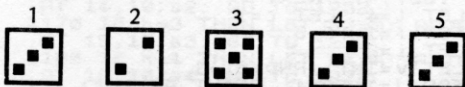
1-ACES	7-3 OF A KIND
2-TWOS	8-4 OF A KIND
3-THREES	9-FULL HOUSE
4-FOURS	10-SMALL STRAIGHT
5-FIVES	11-LARGE STRAIGHT
6-SIXES	12-YAHTZEE
TOTAL:	13-CHANCE
BONUS:	TOTAL:
TOTAL:	GRAND TOTAL:

If you score 63 or more points in section one (the left side), you are awarded a bonus of 35 points.

HOW TO PLAY:

Each player must complete their game before the next one starts (this is done for easier management of the scorecard displays and the actual scoring...and to prevent contestants from playing musical chairs).

The dice are numbered 1 to 5. The first roll is automatic. When the dice change, you are prompted UP TO which number die you want to KEEP. Input the number of the LAST die you wish to keep. For example, if you have rolled:



and you have decided to try for the threes, then you would type: 5 ENTER, because 5 is the number of the LAST die you would want to KEEP. Next you are asked WHICH dice to be kept. For this example you would

type: 1 ENTER, 4 ENTER, 5 ENTER. The number 1, 4 and 5 dice change color to remind you which dice you are saving.

Next you are asked UP TO which number die to THROW. In this case you would type: 3 ENTER. Then you are prompted WHICH dice to throw. Here it would be: 2 ENTER, 3 ENTER and the number 2 and 3 dice will roll again.

IT IS MOST IMPORTANT THAT ALL ABOVE INPUTS BE IN THE PROPER SEQUENCE BECAUSE THE PROGRAM WILL TERMINATE YOUR ROLL WHEN THE LAST NUMBER DIE IS REACHED.

Follow this procedure for 2 more rolls. You can change your mind and re-roll any dice that you were previously saving. If at any time you wish to keep the same dice as a previous roll, or all 5 dice, just enter a ZERO. After one turn (3 rolls) you are asked which category you wish to use. Enter the number of this category. The computer will first calculate the validity of the combination of dice for the category, then the point value (if any), then place that value in the appropriate box on the scorecard and finally tally the totals.

The CHANCE category is used when you get in a jam and have no better options. CHANCE totals all 5 dice values.

When each player finishes their game (13 turns), they are asked if a printout of their final scorecard is desirable. Although the program provides a tally screen of all 4 scores after each player has completed their game, I felt that a screen dump of each scorecard was a nice feature. This works for the 2040 printer only. You will have to alter the code for your particular printer.

THE PROGRAM:

Let's run through the program from the beginning. Line 40 clears memory (50000 to 50008) for the 4 scores. The 2 POKES in line 60 toggle the CAPS LOCK and set the system variable PIP to an audible beep. The DATA statements, lines 720 to 820, hold the graphics for the dice. D\$ is a 3 dimensional array. The 3rd dimension is used for the shadows on the bottom and right sides of the dice.

The "title & table" routine, lines 500 - 700, draws the playing table, the dice and resets the scorecard variables. We then return to the scorecard routine at line 150 which plots out the actual tally sheet.

Player 1 now starts the game in the main loop at line 400. The dice are rolled and control is routed in and out of the "throw dice" and "keep dice" sub-routines for 3 rolls.

After 3 rolls we are directed to the scoring routine which first prompts the player which category is to be used. If this category is within the b1 and b6 variable range (3 OF A KIND to YAHTZEE) program flow is routed to the sorting routine at line 3000. This sorts the values of the dice (first DATA figures in the "dice" statements).

Sorting of this data is necessary to ascertain the validity of the dice thrown and the category chosen. Boolean logic takes over from lines 1117 to 1236. I admit that boolean logic in TS Basic is cumbersome (the ELSE command would be most helpful for this application) but it does work, if somewhat slowly.

Lines 1240 to 1300 perform the tabulations of the scores. When x=14 (line 415), all categories have been used and control goes to the "poke scores" routine at 1380 which displays all 4 scores.

When all 4 games are completed (z>4, line 1420), the "end routine" is accessed where we are given the option of either playing again or quitting.

This is a long listing. If anyone is interested in the program but doesn't feel like keying it in, I will send copies on tape for \$8.00. Also included on the tape will be 4 type fonts which can be loaded into other programs or used while writing your own programs. Full instructions included. Send to: Arthur Marzano, 1949 Locust St., Phila., PA 19103.

VARIABLES

A1 to A6 = ACES - SIXES
 B1 = 3 OF A KIND
 B2 = 4 OF A KIND
 B3 = FULL HOUSE
 B4 = SMALL STRAIGHT
 B5 = LARGE STRAIGHT
 B6 = YAHTZEE
 B7 = CHANCE
 V = Random generator
 L = Line co-ord
 C = Column co-ord
 M = Control for scores
 CC, DD = Control variables for sort routine

T1 = Subtotal, section 1
 T2 = Total+Bonus section 1
 T3 = Total, section 2
 GT = Grand Total
 X = Counter for each turn
 Z = Counter for players
 E(1-5) = Dice values
 D\$ = The dice
 F, I, G, J = Counters
 H = Player input
 K = Control for input

```

10 REM YAHTZEE © 1986
20 REM by Arthur B. Marzano
40 FOR f=50000 TO 50008: POKE
f,0: NEXT f
50 LET q=50000: LET z=1
60 POKE 23658,8: POKE 23609,20
70 BORDER 1: PAPER 1: INK 5: C
LS
80 PRINT INK 6; PAPER 0; BRIGHT
T 0;" YAHTZEE © 1986
6 by Arthur B. Marzano

100 PAUSE 200
110 BORDER 2: INK 0: PAPER 7: O
VER 0: CLS
120 LET l=4: LET c=1
130 GO SUB 710: GO SUB 500
150 REM score card
200 INK 1
210 PRINT AT 13,1,1;" ACES";TAB
15,7;" 3 OF KIND"
220 PRINT TAB 1,2;" TWOS";TAB 1
5,8;" 4 OF KIND"
230 PRINT TAB 1,3;" THREES";TAB
15,9;" FULL HOUSE"
240 PRINT TAB 1,4;" FOURS";TAB
14,10;" SMALL ST"
250 PRINT TAB 1,5;" FIVES";TAB
14,11;" LARGE ST"
260 PRINT TAB 1,6;" SIXES";TAB
14,12;" YAHTZEE"
270 PRINT TAB 3;"TOTAL: ";TAB 1
4,13;" CHANCE"
280 PRINT TAB 3;"BONUS: ";TAB 1
7;"TOTAL: "
290 PRINT TAB 1;"TOTAL: ";TAB 1
4;"GRAND TOTAL: "
300 FOR n=0 TO 79 STEP 8
310 PLOT 2,n: DRAW 250,0
320 NEXT n
330 INK 0

```

```

340 PLOT 2,80: DRAW 250,0: PLOT
2,81: DRAW 250,0: PLOT 2,72: DR
AW 250,0: PLOT 0,0: DRAW 0,81: P
LOT 1,0: DRAW 0,81
380 PLOT 74,0: DRAW 0,71: PLOT
107,0: DRAW 0,71: PLOT 218,0: DR
AW 0,71: PLOT 253,0: DRAW 0,81:
PLOT 254,0: DRAW 0,81: PLOT 255,
0: DRAW 0,163
390 LET x=0
399 REM main loop
400 LET x=x+1
410 IF x=14 THEN PRINT #1;"PRIN
T OUT ? Y/N": PAUSE 0: LET i$=IN
KEY$: IF INKEY$="Y" THEN PRINT P
APER 4;AT 10,1;" ": COPY : LPRI
NT :Z;"
415 IF x=14 THEN PAUSE 300: GO
TO 1380
420 GO SUB 830
430 GO SUB 1030
440 GO SUB 950
450 GO SUB 1030
460 FOR f=1 TO 25: BEEP .008,f:
BEEP .006,-f: NEXT f: PRINT AT
10,1: PAPER 4;" Player ";Z;" 3r
d roll..." : PAUSE 50: GO
SUB 960
470 GO SUB 1080
480 GO TO 400
500 REM title & table
510 PRINT PAPER 2;"
PAPER 6; INK 1;" Y A H T Z E E "
; PAPER 2;"
520 PRINT PAPER 4; INK 0;"
530 FOR f=2 TO 11
540 PRINT PAPER 4;"
550 PRINT PAPER 4;AT 1,0;"
560 FOR f=2 TO 11: PRINT PAPER
4;AT f,0;" : NEXT f
570 LET l=4: LET c=1
580 FOR f=2 TO 5
590 PRINT AT l,c*5;d$(1,f);AT l
,c*5;d$(2,f);AT l,c*5;d$(3,f);
AT l,c*5;d$(4,f);AT l,c*5;d$(5,f)
600 LET l=l+1
610 NEXT f
620 PRINT AT 2,6; PAPER 4;"1";A
T 2,11;"2";AT 2,16;"3";AT 2,21;"
4";AT 2,26;"5"
670 LET a1=0: LET a2=0: LET a3=
0: LET a4=0: LET a5=0: LET a6=0
680 LET bonus=0: LET t1=0: LET
t2=0: LET t3=0: LET gt=0
690 LET b1=0: LET b2=0: LET b3=
0: LET b4=0: LET b5=0: LET b6=0:
LET b7=0
695 DIM e(5): LET e=5: LET c=1
700 RETURN
710 REM dice
720 RESTORE 770
730 DIM d$(6,5,4)
740 FOR f=1 TO 6: FOR i=1 TO 5
750 READ d$(f,i)
760 NEXT i: NEXT f: RETURN
770 DATA "1","1","1","1","1","1"
780 DATA "2","1","1","1","1","1"
790 DATA "3","1","1","1","1","1"
800 DATA "4","1","1","1","1","1"
810 DATA "5","1","1","1","1","1"
820 DATA "6","1","1","1","1","1"
830 REM throw dice
840 INK 0: PAPER 7: PRINT AT 10
,1: PAPER 4;" Player ";Z;" 1st
roll..."
860 FOR f=1 TO 40: BEEP .009,f:
BEEP .007,-f: NEXT f
870 LET l=4: LET c=1
880 FOR f=1 TO 5
890 LET i=2
900 LET v=INT (RND*6)+1
910 FOR j=1 TO 4: PRINT AT l,c*
5;d$(v,i): LET e(c)=VAL d$(v,i):
LET l=l+1: LET i=i+1: NEXT j
920 LET c=c+1: LET l=4
930 NEXT f
940 RETURN
950 FOR f=1 TO 20: BEEP .007,f:
BEEP .005,-f: NEXT f: PRINT AT
10,1: PAPER 4;" Player ";Z;" 2n
d roll..." : PAUSE 50
960 PRINT PAPER 4;AT 10,1;"Up t
o which # die to be thrown": INP
UT h

```



```

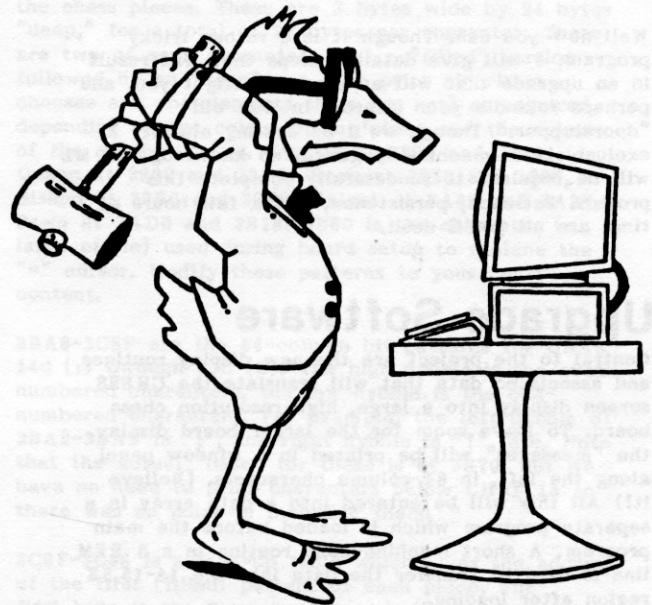
965 IF h>=6 THEN GO TO 960
970 FOR k=1 TO h: PRINT AT 10,1
: PAPER 4;"Which dice to be thro
wn?": INPUT k
975 IF k>=6 THEN GO TO 970
976 LET c=k: LET l=4: LET i=2
1000 LET v=INT (RND*6)+1
1010 FOR j=1 TO 4: PRINT AT l,c*
5;d$(v,i): LET e(c)=VAL D$(v,i):
LET l=l+1: LET i=i+1: NEXT j: N
EXT k
1020 RETURN
1030 REM keep dice
1040 PRINT PAPER 4;AT 10,1;"Up t
o which # die to be kept?": INP
UT h
1045 IF h>=6 THEN GO TO 1040
1050 FOR k=1 TO h: PRINT AT 10,1
: PAPER 4;"Which dice to be kep
t?": INPUT k
1055 IF k>=6 THEN GO TO 1050
1056 GO SUB 1070: LET c=k: NEXT
k
1060 RETURN
1070 FOR f=2 TO 5: PRINT OVER 1;
PAPER 5;AT f+2,k*5;"": NEXT
f: RETURN
1080 REM scoring
1090 FOR f=30 TO 50: BEEP .009,f
: BEEP .005,-f: NEXT f: PRINT AT
10,1; PAPER 4; INK 7;"In which
category?": INPUT h
1095 IF h>=14 THEN GO TO 1090
1100 INK 1
1102 GO SUB 1330
1103 FOR f=1 TO e
1104 IF h=e(c) THEN LET m=m+e(c)
1108 IF h<7 THEN GO TO 1150
1109 IF h=13 THEN GO TO 1230
1110 IF h=7 OR h=8 OR h=9 OR h=1
0 OR h=11 OR h=12 THEN LET e(f)=
e(c): GO SUB 3000
1111 IF h=11 THEN GO TO 1130
1112 IF h=12 THEN GO TO 1140
1113 IF h=7 THEN GO TO 1210
1114 IF h=8 THEN GO TO 1220
1115 IF h=10 THEN GO TO 1120
1117 IF h=9 AND NOT (e(1)=e(2) A
ND e(4)=e(5) AND (e(3)=e(1) OR e
(3)=e(4))) THEN LET b3=0: PRINT
AT 15,28;b3: GO TO 1232
1118 IF h=9 THEN LET b3=25: PRIN
T AT 15,28;b3: GO TO 1232
1120 IF e(1)=e(2) AND e(2)+1=e(3
) AND e(3)+1=e(4) AND e(4)+1=e(5
) THEN LET b4=30
1122 IF e(1)+1=e(2) AND e(2)+1=e
(3) AND e(3)+1=e(4) THEN LET b4=
30
1124 IF e(1)+1=e(2) AND e(2)=e(3
) AND e(3)+1=e(4) AND e(4)+1=e(5
) THEN LET b4=30
1126 IF e(1)+1=e(2) AND e(2)+1=e
(3) AND e(3)=e(4) AND e(4)+1=e(5
) THEN LET b4=30
1128 IF e(1)+1<e(2) AND e(2)+1=e
(3) AND e(3)+1=e(4) AND e(4)+1=e
(5) THEN LET b4=30
1129 PRINT AT 16,28;b4: GO TO 12
32
1130 IF (e(1)<e(2) AND e(2)<e(3)
AND e(3)<e(4) AND e(4)<e(5)) TH
EN LET b5=40
1135 PRINT AT 17,28;b5: GO TO 12
32
1140 LET m=m+e(c): IF (e(1)*5=m)
THEN LET b6=50: PRINT AT 18,28;
b6: GO TO 1232
1145 IF NOT ((e(1)=e(2) AND e(2)
=e(3) AND e(3)=e(4) AND e(4)=e(5
))) THEN LET b6=0: PRINT AT 18,2
8;b6: GO TO 1232
1146 GO TO 1232
1150 IF h=1 THEN LET a1=m: PRINT
AT 13,10;a1: GO TO 1232
1160 IF h=2 THEN LET a2=m: PRINT
AT 14,10;a2: GO TO 1232
1170 IF h=3 THEN LET a3=m: PRINT
AT 15,10;a3: GO TO 1232
1180 IF h=4 THEN LET a4=m: PRINT
AT 16,10;a4: GO TO 1232
1190 IF h=5 THEN LET a5=m: PRINT
AT 17,10;a5: GO TO 1232
1200 IF h=6 THEN LET a6=m: PRINT
AT 18,10;a6: GO TO 1232
1210 IF (e(1)=e(2) AND e(2)=e(3)
) OR (e(2)=e(3) AND e(3)=e(4)) O
R (e(3)=e(4) AND e(4)=e(5)) THEN
LET m=m+e(c): LET b1=m: PRINT A
T 13,28;b1: GO TO 1232
1212 IF NOT b1 THEN PRINT AT 13,
28;b1: GO TO 1232
1220 IF (e(1)=e(2) AND e(2)=e(3)
AND e(3)=e(4)) OR (e(2)=e(3) AN
D e(3)=e(4) AND e(4)=e(5)) THEN
LET m=m+e(c): LET b2=m: PRINT AT
14,28;b2: GO TO 1232

```

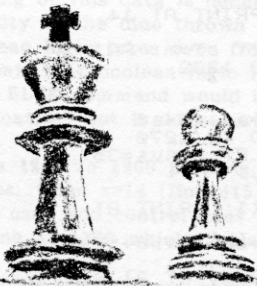
```

1225 IF NOT b2 THEN PRINT AT 14,
28;b2: GO TO 1232
1230 LET m=m+e(c): LET b7=m: PRI
NT AT 19,28;b7: GO TO 1232
1235 LET c=c+1
1236 NEXT f
1240 LET t1=a1+a2+a3+a4+a5+a6
1250 IF t1<63 THEN GO TO 1270
1260 IF t1>=63 THEN LET bonus=35
: PRINT AT 20,10;bonus
1270 LET t2=t1+bonus
1280 PRINT AT 19,10;t1: PRINT AT
21,10;t2
1290 LET t3=b1+b2+b3+b4+b5+b6+b7
: PRINT AT 20,28;t3
1300 LET gt=t2+t3: PRINT AT 21,2
8;gt
1320 RETURN
1330 LET c=1
1340 LET m=0
1370 RETURN
1375 REM poke scores 1 - 4
1380 INK 0: PAPER 7: POKE (q+z),
gt-256*INT (gt/256): POKE (q+z+1
),INT (gt/256): CLS
1390 INK 0: OVER 0
1395 PRINT : PRINT
1400 PRINT "TAB 9;"Player 1 - "
: PRINT PEEK 50001+256*PEEK 500
02: PRINT "TAB 9;"Player 2 - "
: PRINT PEEK 50003+256*PEEK 500
04: PRINT "TAB 9;"Player 3 - "
: PRINT PEEK 50005+256*PEEK 500
06: PRINT "TAB 9;"Player 4 - "
: PRINT PEEK 50007+256*PEEK 500
08
1410 LET q=q+1
1420 PAUSE 300: LET z=z+1: IF z>
4 THEN GO TO 4000
1430 CLS: GO TO 110
2999 REM sort routine
3000 FOR i=1 TO e-1
3010 FOR g=1 TO e-i
3020 LET cc=e(g)
3030 LET dd=e(g+1)
3040 IF cc<dd THEN GO TO 3070
3050 LET e(g)=dd
3060 LET e(g+1)=cc
3070 NEXT g
3080 NEXT i
3090 RETURN
3999 REM end routine
4000 PRINT AT 20,0; INK 1; FLASH
1;"": P L A Y A G A I N Y/N
?
4010 PAUSE 0
4020 IF INKEY$="Y" THEN RUN
4030 BORDER 1: PAPER 1: INK 5: C
LS: PRINT AT 10,0; FLASH 1; BRI
GHT 1;"": THANKS FOR PLAYING -
BYE
9997 STOP
9998 SAVE "YAHTZEE" LINE 10

```



"HIT ANY KEY TO CONTINUE"



Upgrade to Hi-res CHESS

by Fred Nachbaur

Courtesy Silicon Mountain Computers

Most of you are undoubtedly as thrilled as I am, about the WRX16 high resolution core routines published in SyncWare News #4:2. This software opens new vistas (quite literally) for our ZX81's and TS1000/1500's. In fact, several new programs are already available from Silicon Mountain, including an Extended BASIC, a dungeon game, and various utilities. The only hardware requirement is a suitable 8K NVM board as a "Hunter" board or our own updated version, or even the built-in NVM published in SWN 4:1.

What about existing software? Can any of it be easily upgraded to a high resolution format? The answer is a resounding YES. BASIC programs, of course, can be upgraded using SRAM HI*RES. But what about some of the great machine-code programs that have come along over the years? Once again, the answer is YES, although such projects will demand more ability and/or ingenuity.

One of my all-time favourite programs for the ZX81 family is Psion's ZX CHESS, as distributed by Timex. (I actually prefer it over Psion's QL CHESS, since I'm not that good a chess player.) Its only real drawback is the inadequacy of the board display; it is a virtual necessity to "play along" using a real board. Still, it is remarkably competent for a program that runs in just over 8K of memory. Wouldn't it be great if you could play this with "real" chess pieces on the screen?

Well now you can. Though it is a rather tricky program, I will give detailed steps that will result in an upgrade that will amaze and delight you, and perhaps rekindle your interest in that old "doorstopper." Though we'll be dealing almost exclusively in machine language, no knowledge of ML will be required to successfully complete this project. Patience, persistence, and a few hours of time are all you'll need.

Upgrade Software

Central to the project are the new display routines and associated data that will translate the CHESS screen display into a large, high-resolution chess board. To leave room for the larger board display, the "messages" will be printed in a window panel along the left, in 64-column characters. (Believe it!) All this will be entered into a data array in a separate program which is loaded before the main program. A short machine-code routine in a 0 REM line is used to transfer the data into the 14-15.5K region after loading.

Get Started

Start by powering up your computer with 16K RAM plus suitable 8K NVM. Enter the loader program of LISTING 1, and save it to tape just in case Murphy strikes.

Note that line 1 is a REM followed by 28 characters. Enter POKE 16510,0; this gives the REM a line number of 0, making it un-deletable. Then enter RUN.

Enter the Code

Now take a deep breath, get good and comfortable, and start entering the data and machine code of TABLE 1. The data is arranged in "screens" of hex numbers, each with a specific start address. Enter the start address, then the hex numbers. Go from left to right, top to bottom. You'll be entering over 1600 hex numbers, so take your time. Save to tape frequently, preferably after every "screen," by entering GOTO 100. Take a break whenever you start feeling error-prone. Finding errors "after the fact" is MUCH harder than being careful in the first place. The data is stored in string array A\$, so you should be aware NEVER to use RUN or CLEAR after you've started entering data. Restart after every screen with GOTO 10.

There are quite a few zeros (00) in the data tables. To make things a little easier for you, the loader will accept ENTER (only) to mean zero.

Downloader

When all the data has been entered, delete the loader lines (all that will be left is line 0 REM), and enter the "REM-loader" of LISTING 2. GOTO 10, and enter the values of TABLE 2. This is a little down-loader that will transfer the data in A\$ into its operating region from 14-15.5K.

Once you're all done, delete the REM-loader program lines. In their place, enter the "boot" program of LISTING 3. SAVE it to a fresh, clean tape with GOTO 10. After the SAVE, and subsequently LOAD, the data will be transferred into low memory. Don't power down (or if you do, reload the program before proceeding), as some of the routines are needed for the next step.

Modify 'CHESS'

Now that the laborious part is complete, we only have to modify the CHESS program so that it will work with the new software. Follow the instructions EXACTLY, and in the order that they are given; as I implied earlier, this is not the kind of program you'll want to get "creative" with unless you know what you're doing.

LOAD "CHESS". Do not use the RAND USR 837 trick for "breaking" the program. When loading is complete, the screen will (presumably) say, "LOAD CHECK PASSED. WOULD YOU LIKE A GAME OF CHESS?" Press EDIT (shift 1), which will remove the quotes from the input cursor. Now enter STOP, and you'll get report D/3.

Now enter the following immediate commands (no line numbers):

```
DIM B$(37)
LET B$( TO 2)=CHR$ 237+CHR$ 79
LET B$(35 TO 9)=CHR$ 195+CHR$ 159+CHR$ 60
```

If you prefer the message window in inverse video (I do), then also enter the following command:

```
LET B$(3 TO 9)="      " (7 inverse spaces)
```

Now the following:

```
LET A=15572
LET B=15578
LET SAVE=15906 (SAVE is spelled out)
```

Time to do a check on your work. Enter PRINT PEEK 24707. This should print 237. If it doesn't, you did something wrong.

Now enter the following program lines (with line numbers), overwriting the lines in the BASIC driver. Enter them EXACTLY as shown; this is critical.

```
4 LET T=USR A
5 LET T=USR B
6 GOTO 3
```

Again check PRINT PEEK 24707. Again you should get 237.

Now we'll enter the following list of POKes (no line numbers). These make cosmetic changes to the various messages, to allow them to fit into the narrower window:

```
POKE 17499,118
POKE 17522,118
POKE 18042,118
POKE 23866,28
POKE 17813,118
POKE 17821,39
POKE 17822,52
POKE 17823,38
POKE 17824,55
POKE 17825,41
POKE 17826,15
POKE 17844,118
POKE 24493,118
POKE 24504,0
POKE 24505,29
POKE 24506,22
POKE 24507,34
POKE 24508,0
POKE 24519,118 (NOTE: 24519, not 24509)
```

(optional: executes NEW instead of reset when quitting, i.e. preserves RAMTOP.)

```
POKE 23879,195
POKE 23880,3
```

SAVING

There! All done! Well, almost; we still have to save it to tape. You probably know that you can't just GOTO 1, for protection purposes. So we have to use a little trickery to get around this. Start your tape a few seconds after the end of the "boot" program, and enter GOTO USR SAVE. (GOTO and USR are the tokens, but SAVE is spelled out.) The program will save to tape, while keeping the LOAD CHECK routines active to warn of subsequent bad loads, and bypassing the crash-and-burn anti-piracy feature.

To make backups of the boot program, ("HRCHESS"), break operation when loaded and GOTO 10. To make a backup of the modified "CHESS" program, press EDIT then enter STOP after loading, then GOTO USR SAVE. (The HRCHESS boot program must be present.)

RUNNING

To load the high-res chess program, enter LOAD "HRCHESS" which, when loaded, will automatically chain the modified "CHESS" portion. The very first screen will be pretty much as you're accustomed to; the subsequent displays, however, are like nothing you've seen on a ZX81 before. But don't take my word for it, see for yourself! Incidentally, this will run on a TS1500 if you add a 16K RAM pack in addition to the SRAM.

Since the new display takes 26 lines, the picture might spill off the top or bottom of your TV. Simply adjust the VERT. HEIGHT control on the back to bring it on-screen if necessary.

If you want a graphic demonstration of the difference the new display makes, or if you have a nostalgic urge to view the old display, hold down the SHIFT key while the computer is thinking about its next move. This will show the subsequent move in the original "stone-age" mode.

If you don't want to go to the trouble of entering all the machine-code, or if your attempts fail, I will supply tapes of the "HRCHESS" boot program only (Psion CHESS is copyrighted, so I can't supply the modified program unless you send your original tape). Send \$10 to Silicon Mountain Computers, C-12 Mtn. Stn. Group Box, Nelson, BC V1L 5P1 Canada. Also write to inquire about our other high-res products, or our "SCRAM" 8K NVM board which will support this software.

Tech Info

The high-res display file is mapped at 2000-37FFh. The data from 3800h to 3BA7 are the bit patterns for the chess pieces. These are 3 bytes wide by 24 bytes "deep," for a total of 72 bytes per character. There are two of each character, first a "filled" version followed by an "open" one. The print algorithm chooses and complements them (or not) as required, depending on the colour of the piece and the colour of the square it's on. King is at 3800 and 3848, Queen at 3890 and 38D8, Rook at 3920 and 3968, Bishop at 39B0 and 39F8, Knight at 3A40 and 3A88, Pawn at 3AD0 and 3B18. 3B60 is the character (a large circle) used during board setup to replace the "=" cursor. Modify these patterns to your heart's content.

3BA8-3C6F are the 64-column bit patterns for CHR\$ 14d (:) through 63d (Z). The high nybble is the even-numbered characters, the low nybble is the odd-numbered characters. For example, the left nybble of 3BA8-3BAF is ":", the right nybble is "?", etc. Note that the actual "base" for these is at 3B70, but we have no need to print characters below CHR\$ 15 so there was no point in including them.

3C6F-3C86 is a look-up table which gives the address of the first (filled) pattern of each piece. The first byte is the Sinclair character (K, Q, R, etc.), the next two are the address of the pattern.

3C88-3E62 is all code. 3C88-3CD3 is the WRX16 core. 3CD4 is where we return to FAST mode, thence to the program.

3CDA-3E21 is our display routine. This code segment is relocatable, i.e. no JPs or self-referred CALLs. At 3CDA we check for "shift," returning if pressed. 3CE1 makes sure that the dummy display file exists in high memory (making it usable with 64K, provided that its 8-16K region is disabled and SRAM used instead). 3CEC clears the high-res display file. 3CF9 copies the messages from the normal display file to the 64-column window. (Want a challenge? Figure THAT one out!)

3D7F prints the board, using the contents of the normal display file as a template. This is another fairly involved routine. 3DF5 prints the frame around the board. Finally, 3E1A calls SLOW mode and sets the display vector (IX) to enter high-res mode. (As a side note, this entire stretch of code takes less than a second to run, even with all those nested loops!)

The last routine is the USR SAVE program at 3E22. This computes the checksum and stores it in the reference bytes at 6072, and undoes Psion's "auto-crash" at 6074.

The area from 3E70 to 3EFF is filled with zeros by the downloader; this represents the "pattern" for zero (blank squares), and should not be used for other purposes. 3FF0-3FF2 are variables used by the board-printing routine. The bytes from 3F00 to 3FEF and 3FF3 to 3FFF are yours to do with as you see fit.

LISTING #1

```

1 REM XXXXXXXXXXXXXXXXXXXXXXXX
XXXXX
10 DIM A$(1648)
20 CLS
30 PRINT "ADDRESS=";
35 INPUT AD
40 CLS
45 PRINT "ADDRESS=";AD
50 LET SUM=0
55 LET A=AD
60 FOR L=1 TO 20
70 FOR C=0 TO 30 STEP 3
80 INPUT H$
90 IF H$="" THEN LET H$="00"
100 IF LEN H$<>2 THEN GOTO 80
110 LET N=16*CODE H$(1)+CODE H$
(2)-476
120 IF N>255 OR N<0 THEN GOTO 8
0
130 PRINT AT L,C;H$
140 LET A$(A-14335)=CHR$ N
150 LET SUM=SUM+N
160 LET A=A+1
170 IF A=15984 THEN GOTO 200
180 NEXT C
190 NEXT L
200 PRINT AT 21,0;"CHECKSUM=";S
UM;" OK? Y/N"
210 PAUSE 4E4
220 IF INKEY$="Y" THEN GOTO 250
230 IF INKEY$="N" THEN GOTO 40
240 GOTO 210
250 CLS
260 PRINT "SAVE? Y/N"
270 PAUSE 4E4
280 IF INKEY$="Y" THEN SAVE "TA
BLE1"
290 IF INKEY$<>"N" THEN GOTO 25
0
300 GOTO 20

```

LISTING #2

```

0 REM XXXXXXXXXXXXXXXXXXXXXXXX
XXXXX
10 FOR A=16514 TO 16541
20 INPUT N
30 LET N$=STR$ N+" "
40 PRINT N$( TO 3);":";
50 POKE A,N
60 NEXT A

```

LISTING #3

```

10 REM HRCHESS BOOT PROGRAM
20 SAVE "HRCHESS"
30 PRINT "****DO NOT TURN OFF
RECORDER****STAND BY TO LOAD HIG
HRES "CHESS""
40 RAND USR 16516
50 PAUSE 300
60 LOAD "CHESS"

```

TABLE 1

ADDRESS=14336

```

00 00 00 00 10 00 00 10 00 00 7C
00 00 10 00 00 10 00 00 38 00 00
7C 00 00 7C 00 00 7C 00 00 38 00
00 28 00 00 7C 00 00 FE 00 00 FE
00 01 FF 00 01 FF 00 00 FE 00 01
FF 00 01 FF 00 03 FF 80 06 AA C0
07 FF C0 00 00 00 00 00 00 00 10
00 00 10 00 00 7C 00 00 10 00 00
10 00 00 38 00 00 44 00 00 44 00
00 44 00 00 28 00 00 38 00 00 6C
00 00 C6 00 00 82 00 01 01 00 01
01 00 00 82 00 01 83 00 01 01 00
03 FF 80 06 AA C0 07 FF C0 00 00
00 00 00 00 00 10 00 01 7D 00 00
FE 00 00 FE 00 00 7C 00 00 38 00
00 28 00 00 7C 00 00 FE 00 00 FE
00 01 FF 00 01 FF 00 01 FF 00 00
FE 00 00 FE 00 00 7C 00 00 54 00
00 FE 00 01 FF 00 03 FF 80 06 AA
C0 07 FF C0 00 00 00 00 00 00 00
CHECKSUM=10990

```

ADDRESS=14556

```

10 00 01 7D 00 00 C6 00 00 82 00
00 44 00 00 28 00 00 38 00 00 44
00 00 C6 00 00 82 00 01 01 00 01
01 00 01 01 00 00 82 00 00 C6 00
00 44 00 00 54 00 00 82 00 01 01
00 03 FF 80 06 AA C0 07 FF C0 00
00 00 00 00 00 00 00 00 00 00
00 00 00 01 99 80 01 99 80 01 FF
80 01 24 80 01 FF 80 00 FF 00 00
7E 00 00 7E 00 00 7E 00 00 3C 00
00 3C 00 00 24 00 00 7E 00 00 FF
00 00 FF 00 01 FF 80 01 99 80 03
FF C0 03 FF C0 00 00 00 00 00 00
00 00 00 00 00 00 00 00 01 99
80 01 99 80 01 FF 80 01 00 80 01
81 80 00 C3 00 00 42 00 00 42 00
00 42 00 00 24 00 00 24 00 00 3C
00 00 66 00 00 81 00 00 81 00 01
00 80 01 66 80 03 00 C0 03 FF C0
00 00 00 00 00 00 00 00 00 10
CHECKSUM=10585

```


ADDRESS=14776

00 00 10 00 00 38 00 00 74 00 00
6C 00 00 7C 00 00 7C 00 00 38 00
00 28 00 00 7C 00 00 7C 00 00 FE
00 00 FE 00 00 FE 00 00 FE 00 00
7C 00 00 7C 00 00 FE 00 01 FF 00
03 55 80 03 FF 80 00 00 00 00 00
00 00 00 00 00 10 00 00 10 00 00
28 00 00 4C 00 00 54 00 00 44 00
00 6C 00 00 28 00 00 38 00 00 6C
00 00 44 00 00 82 00 00 82 00 00
82 00 00 C6 00 00 44 00 00 44 00
00 82 00 01 FF 00 03 55 80 03 FF
80 00 00 00 00 00 00 00 80 00 00
C0 00 00 70 00 01 FC 00 01 E4 00
01 FF 00 01 FF 80 00 FF 80 00 FC
00 00 78 00 00 38 00 00 38 00 00
7C 00 00 FE 00 00 FE 00 00 FE 00
00 7C 00 00 7C 00 00 FE 00 01 FF
00 03 55 80 03 FF 80 00 00 00 00
00 00 00 80 00 00 C0 00 00 F0 00
CHECKSUM=10883

ADDRESS=14996

01 0C 00 01 34 00 01 03 00 01 00
80 00 83 80 00 C4 00 00 68 00 00
28 00 00 28 00 00 44 00 00 82 00
00 82 00 00 C6 00 00 44 00 00 44
00 00 82 00 01 FF 00 03 55 80 03
FF 80 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 38 00 00
7C 00 00 FE 00 01 FF 00 01 FF 00
01 FF 00 00 FE 00 00 7C 00 00 38
00 00 38 00 00 38 00 00 7C 00 01
FF 00 03 FF 80 03 FF 80 00 00 00
00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00
00 00 00 38 00 00 44 00 00 82 00
01 01 00 01 01 00 01 01 00 00 82
00 00 44 00 00 28 00 00 28 00 00
28 00 00 44 00 00 82 00 03 83 80
03 FF 80 00 00 00 00 00 00 00 00
00 00 00 00 00 FF 00 03 00 C0 04
CHECKSUM=7266

ADDRESS=15216

00 20 08 00 10 08 00 10 10 00 08
10 00 08 10 00 08 10 00 08 10 00
08 10 00 08 10 00 08 10 00 08 08
00 10 08 00 10 04 00 20 03 00 C0
00 FF 00 00 00 00 00 00 00 00 00
00 00 04 4A 02 04 40 04 00 00 24
42 42 42 42 24 00 00 82 44 28 44
82 00 00 00 04 E4 0E E4 04 00 00
00 0A 04 EE 04 0A 00 00 00 20 24
40 40 84 84 08 00 00 00 00 00 4C
4C 80 00 64 AC A4 A4 CE 00 00
4E A2 24 42 8A E4 00 00 2E 68 AC
E2 2A 24 00 00 4E 82 C2 A4 A8 48
00 00 44 AA 4A A6 A2 44 00 00 4C
AA AC EA AA AC 00 00 6C 8A 8A 8A
8A 6C 00 00 EE 88 CC 88 88 E8 00
00 4A AA 8E AA AA 6A 00 00 E6 42
42 42 4A E4 00 00 A8 A8 C8 C8 A8
AE 00 00 AC EA AA AA AA 00 00
4C AA AA AC A8 48 00 00 4C AA AA
CHECKSUM=14918

ADDRESS=15436

AC AA 4A 20 00 6E 84 44 24 24 C4
00 00 AA AA AA AA A4 64 00 00 AA
AA A4 A4 EA AA 00 00 AE A2 44 44
48 4E 00 70 3E 30 00 38 36 90 38
37 20 39 27 B0 39 33 40 3A 2E D0
3A 14 60 3B 00 F3 3E 07 47 00 10
FD C6 EE 3C 20 FD 06 C0 11 20 00
21 00 20 C3 A6 3C 00 00 05 CA AD
3C 19 7C ED 47 7D C3 83 E0 2A 0C
40 2A 0C 40 2A 0C 40 00 11 F7 82
19 3E 1E ED 47 3E F5 01 07 02 CD
B5 02 CD 92 02 CD 20 02 DD 21 88
3C C3 A4 02 CD 23 0F C3 00 4F CD
BB 02 7C FE FE C8 21 83 60 11 83
E0 01 25 00 ED B0 21 00 20 36 00
54 5D 13 01 FF 17 ED B0 2A 0C 40
11 C3 02 19 06 16 C5 0E 0E C5 05
0D 7E A7 28 45 CB 47 1F 37 3F 57
79 1F 4F 7A E5 F5 21 70 3B 16 00
5F CB 23 CB 12 CB 23 CB 12 CB 23
CHECKSUM=20177

ADDRESS=15656

CB 12 19 EB 21 00 20 09 F1 06 08
C5 F5 1A 46 20 36 38 26 E6 F0 F5
78 E6 0F 47 F1 B0 77 13 D5 11 20
00 19 D1 F1 C1 10 E3 E1 C1 2B 0D
20 AF C1 AF 11 13 00 ED 52 10 A3
18 1E CB 3F CB 3F CB 3F CB 3F F5
78 E6 F0 18 D2 38 0A CB 27 CB 27
CB 27 CB 27 18 C2 E6 0F 18 EA 2A
0C 40 11 95 00 19 06 00 C5 0E 00
C5 22 F0 3F 78 81 CB 47 7E 07 1F
CB BF F5 21 6F 3C BE 28 05 23 23
23 18 F8 23 5E 23 56 F1 F5 30 04
20 0D 18 02 28 09 FE 14 28 05 21
48 00 19 EB 78 80 80 47 79 81 81
4F 21 08 20 09 06 18 0E 03 F1 F5
1A 28 01 2F 77 23 13 0D 20 F4 D5
11 1D 00 19 D1 10 EA F1 2A F0 3F
23 23 C1 3E 08 0C B9 20 9F 11 32
00 19 C1 04 B8 20 93 21 08 20 11
E0 17 06 18 36 FF E5 19 36 FF E1
CHECKSUM=20988

ADDRESS=15876

23 10 F6 21 08 20 06 C0 CB FE 11
17 00 19 CB C6 11 09 00 19 10 F2
CD 2B 0F DD 21 88 3C C9 CD 23 0F
CD 2A 0A 11 70 46 3E 0B CD 64 5F
21 7D 40 11 00 00 4E 06 08 CB 39
CB 13 CB 12 30 04 7B EE 2D 5F 10
F2 23 7C FE 60 20 E9 7D FE 6F 20
E4 ED 53 72 60 21 00 46 22 75 60
CD 2A 0A 01 01 00 C9 00 00 00 00
00 00 00 00 00 00 00 00 00 00
CHECKSUM=8607

TABLE 2

118:118:42 :16 :64 :17 :6 :0 :
25 :17 :0 :56 :1 :111:6 :237:
176:98 :107:54 :0 :19 :1 :145:
0 :237:176:201:

CLASSIFIED

Do you have something to sell?
Let everyone know! Just \$2.75 a
line will put it here. 32 column
(screen) format, include spaces
in your text, and send CK or MO
with your listing.

Send to SWN Classified, 602 S.
Mill St., Louisville, OH 44641

HARDWARE HACKERS look at THIS!
Here's a rare find of expansion
goodies for your ZX81/TS1000.

FEMALE EDGE CONNECTORS: Brand
new, WITH slot pin...\$4 each.

DELUXE FEED-THRU CABLE ASSY:
hooded enclosure, male feed
thru, lines brought out to side
in multi-conductor cable.
Ferrite noise reducer. VERY
NICE!.....\$10 each.

Please add \$1.50 postage.

Thomas B. Woods, Box 64,
Jefferson, NH 03583

Still using only 16K of TS1000 RAM? Now cheaply double or triple your memory and unleash your computer's true power.

Zebra Systems has purchased a large lot of new
Memotech 16K and 32K RAMs directly from England so that
we could offer them at drastically reduced prices.

Memotech RAMs are the highest quality and
best-selling expansion RAMs for Timex/Sinclair computers.
They have sleek looking black extruded-aluminum cases
that conform to the shape of your TS1000 or ZX81. These
16K and 32K RAMs feature feed-through connectors and
circuitry that allows you to use them along with the 16K
RAMs you already have, so that you can achieve
respectively, 32K or 48K total system RAM capacity.

Each RAM comes with an easy to understand
instruction manual explaining how to use the RAM to best
advantage, and a 15-day money-back guarantee from
Zebra Systems. Order yours now, while the supply lasts.

Achieve 32K with
Memotech 16K
+ your Timex 16K

or

Achieve 48K with
Memotech 32K
+ your Timex 16K

CAT# R416 Memotech 16K RAM ... \$14.95

CAT# R432 Memotech 32K RAM ... \$24.95

CAT#S116 Sinclair 16K RAM, TS1016 equivalent... \$10.00

Include \$3.00 S&H. VISA/MC accepted. NY residents add sales tax.

ZEBRA SYSTEMS, INC.

78-06 JAMAICA AVE. WOODHAVEN, NY 11421, (718) 296-2385

SyncWare News

CONTENTS

For Your Support...	3
Forum...	4
Soapbox...	8
W.C. McGrath	
Put Your INKJET in	
PLOTTER Mode...	9
Cook	
Quick Print Pro/File &	
Tasword Files...	13
Hartung	
TS1000 Mystery Program...	15
Harder	
Yahtzee 2068...	17
Marzano	
Upgrade to Hi-Res CHESS...	20
Nachbaur (TS1000)	

FIRST-CLASS MAIL
U.S. POSTAGE
PAID
Jefferson, NH
Permit No. 1

H.H. Armstrong
5301 Porter Ranch Rd
Garden Valley, CA

95633

5:2

Editor:
Jeffrey Moore
602 S. Mill Street
Louisville, OH 44641

Technical Director:
Fred Nachbaur
C-12, Mtn. Stn. Group Box
Nelson, BC V1L 5P1 Canada

Advertising, Subscriptions:
Thomas B. Woods
P.O. Box 64
Jefferson, NH 03583

Submissions, announcements, letters to the editor,
and all other material of editorial interest should
be sent to the Ohio address.

Copyright 1986 SyncWare News

Subscribe ! \$ 16.95

For Canada and Mexico please add \$3.00 postage.

1 year (6 issues)